



US009213684B2

(12) **United States Patent**
Lai et al.

(10) **Patent No.:** **US 9,213,684 B2**
(45) **Date of Patent:** **Dec. 15, 2015**

(54) **SYSTEM AND METHOD FOR RENDERING DOCUMENT IN WEB BROWSER OR MOBILE DEVICE REGARDLESS OF THIRD-PARTY PLUG-IN SOFTWARE**

(71) Applicant: **Box, Inc.**, Los Altos, CA (US)

(72) Inventors: **Peter Lai**, Los Altos, CA (US); **Anurag Pandit**, Mountain View, CA (US); **Ryan Damico**, Los Altos, CA (US); **Nick Rolph**, Los Altos, CA (US); **Sean Rose**, San Francisco, CA (US); **Joseph Chung**, Los Altos, CA (US); **Jonathan Matthew Long**, San Francisco, CA (US); **Bennett Lee Rogers**, San Francisco, CA (US)

(73) Assignee: **Box, Inc.**, Los Altos, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 190 days.

(21) Appl. No.: **14/027,147**

(22) Filed: **Sep. 13, 2013**

(65) **Prior Publication Data**
US 2015/0082148 A1 Mar. 19, 2015

(51) **Int. Cl.**
G06F 3/00 (2006.01)
G06F 17/22 (2006.01)
(Continued)

(52) **U.S. Cl.**
CPC **G06F 17/2247** (2013.01); **G06F 17/214** (2013.01); **G06F 17/2264** (2013.01); **G06F 17/241** (2013.01)

(58) **Field of Classification Search**
CPC G06F 17/214
USPC 715/716
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,799,320 A 8/1998 Klug
5,848,415 A 12/1998 Guck
5,864,870 A 1/1999 Guck

(Continued)

FOREIGN PATENT DOCUMENTS

CA 2724521 11/2009
CN 101997924 A 3/2011

(Continued)

OTHER PUBLICATIONS

“Conceptboard”, One-Step Solution for Online Collaboration, retrieved from websites <http://conceptboard.com> and <https://www.youtube.com/user/ConceptboardApp?feature=watch>, printed on Jun. 13, 2013, 9 pages.

(Continued)

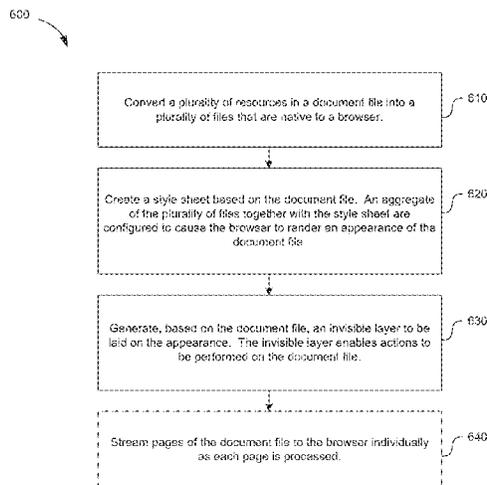
Primary Examiner — William Titcomb

(74) *Attorney, Agent, or Firm* — Perkins Coie LLP

(57) **ABSTRACT**

Techniques are disclosed for render documents that are in proprietary formats in a browser or mobile device. In one embodiment, a method comprises converting a plurality of resources in a document file into a plurality of files that are native to a browser. The method further comprises creating a style sheet based on the document file, wherein an aggregate of the plurality of files together with the style sheet are configured to cause the browser to render an appearance of the document file. The method further comprises generating, based on the document file, an invisible layer to be laid on the appearance, wherein the invisible layer enables actions to be performed on the document file. Among other advantages, embodiments disclosed herein provide the convenience of viewing and accessing documents regardless of whether a software or plug-in therefor is installed.

30 Claims, 9 Drawing Sheets



(56)

References Cited

U.S. PATENT DOCUMENTS

2007/0016680	A1	1/2007	Burd et al.	2009/0249224	A1	10/2009	Davis et al.
2007/0038934	A1	2/2007	Fellman	2009/0254589	A1	10/2009	Nair et al.
2007/0100830	A1	5/2007	Beedubail et al.	2009/0260060	A1*	10/2009	Smith et al. 726/3
2007/0115845	A1	5/2007	Hochwarth et al.	2009/0271708	A1	10/2009	Peters et al.
2007/0118598	A1	5/2007	Bedi et al.	2009/0282212	A1	11/2009	Peterson
2007/0124460	A1	5/2007	McMullen et al.	2009/0300356	A1	12/2009	Crandell
2007/0124737	A1	5/2007	Wensley et al.	2009/0300527	A1	12/2009	Malcolm et al.
2007/0124781	A1	5/2007	Casey et al.	2009/0327358	A1	12/2009	Lukiyanov et al.
2007/0126635	A1	6/2007	Houri	2009/0327961	A1	12/2009	De Vorchik et al.
2007/0130143	A1	6/2007	Zhang et al.	2010/0011292	A1	1/2010	Marinkovich et al.
2007/0130163	A1	6/2007	Perez et al.	2010/0011447	A1	1/2010	Jothimani
2007/0198609	A1	8/2007	Black et al.	2010/0017262	A1	1/2010	Iyer et al.
2007/0208878	A1	9/2007	Barnes-Leon et al.	2010/0036929	A1	2/2010	Scherpa et al.
2007/0214180	A1	9/2007	Crawford	2010/0042720	A1	2/2010	Stienhans et al.
2007/0220016	A1	9/2007	Estrada et al.	2010/0057560	A1	3/2010	Skudlark et al.
2007/0220590	A1	9/2007	Rasmussen et al.	2010/0057785	A1	3/2010	Khosravy et al.
2007/0240057	A1	10/2007	Satterfield et al.	2010/0076946	A1	3/2010	Barker et al.
2007/0256065	A1	11/2007	Heishi et al.	2010/0082634	A1	4/2010	Leban
2007/0266304	A1	11/2007	Fletcher et al.	2010/0088150	A1	4/2010	Mazhar et al.
2007/0282848	A1	12/2007	Kiilerich et al.	2010/0092126	A1	4/2010	Kaliszek et al.
2007/0283443	A1	12/2007	McPherson et al.	2010/0093310	A1	4/2010	Gbadegesin et al.
2007/0288290	A1	12/2007	Motoyama et al.	2010/0107225	A1	4/2010	Spencer et al.
2008/0005135	A1	1/2008	Muthukrishnan et al.	2010/0131868	A1	5/2010	Chawla et al.
2008/0005195	A1	1/2008	Li	2010/0151431	A1	6/2010	Miller
2008/0016146	A1	1/2008	Gan et al.	2010/0153835	A1	6/2010	Xiong et al.
2008/0028323	A1	1/2008	Rosen et al.	2010/0162365	A1	6/2010	Del Real
2008/0040173	A1	2/2008	Aleong et al.	2010/0162374	A1	6/2010	Nair
2008/0046828	A1	2/2008	Bibliowicz et al.	2010/0179940	A1	7/2010	Gilder et al.
2008/0059656	A1	3/2008	Saliba et al.	2010/0185463	A1	7/2010	Noland et al.
2008/0063210	A1	3/2008	Goodman et al.	2010/0185932	A1	7/2010	Coffman et al.
2008/0065881	A1	3/2008	Dawson et al.	2010/0191689	A1	7/2010	Cortes et al.
2008/0077631	A1	3/2008	Petri	2010/0198783	A1	8/2010	Wang et al.
2008/0091763	A1	4/2008	Devonshire et al.	2010/0198871	A1	8/2010	Stiegler et al.
2008/0091790	A1	4/2008	Beck	2010/0198944	A1	8/2010	Ho et al.
2008/0104277	A1	5/2008	Tian	2010/0205537	A1	8/2010	Knighton et al.
2008/0114720	A1	5/2008	Smith et al.	2010/0223378	A1	9/2010	Wei
2008/0133674	A1	6/2008	Knauerhase et al.	2010/0229085	A1	9/2010	Nelson et al.
2008/0140732	A1	6/2008	Wilson et al.	2010/0235526	A1	9/2010	Carter et al.
2008/0147790	A1	6/2008	Malaney et al.	2010/0235539	A1	9/2010	Carter et al.
2008/0151817	A1	6/2008	Fitchett et al.	2010/0241611	A1	9/2010	Zuber
2008/0154873	A1	6/2008	Redlich et al.	2010/0241972	A1	9/2010	Spataro et al.
2008/0182628	A1	7/2008	Lee et al.	2010/0250120	A1	9/2010	Waupotitsch et al.
2008/0183467	A1	7/2008	Yuan et al.	2010/0251340	A1	9/2010	Martin et al.
2008/0184130	A1	7/2008	Tien et al.	2010/0257457	A1	10/2010	De Goes
2008/0194239	A1	8/2008	Hagan	2010/0262582	A1	10/2010	Garcia-Ascanio et al.
2008/0215883	A1	9/2008	Fok et al.	2010/0267588	A1	10/2010	Nelson et al.
2008/0222654	A1	9/2008	Xu et al.	2010/0274765	A1	10/2010	Murphy et al.
2008/0243855	A1	10/2008	Prahlad et al.	2010/0274772	A1	10/2010	Samuels
2008/0250333	A1	10/2008	Reeves et al.	2010/0281118	A1	11/2010	Donahue et al.
2008/0250348	A1	10/2008	Alimpich et al.	2010/0290623	A1	11/2010	Banks et al.
2008/0263099	A1	10/2008	Brady-Kalnay et al.	2010/0306379	A1	12/2010	Ferris
2008/0271095	A1	10/2008	Shaffon	2010/0318893	A1	12/2010	Matthews et al.
2008/0276158	A1	11/2008	Lim et al.	2010/0322252	A1	12/2010	Suganthi et al.
2009/0015864	A1	1/2009	Hasegawa	2010/0325155	A1	12/2010	Skinner et al.
2009/0019093	A1	1/2009	Brodersen et al.	2010/0325527	A1	12/2010	Estrada et al.
2009/0019426	A1	1/2009	Baeumer et al.	2010/0325559	A1	12/2010	Westerinen et al.
2009/0030710	A1	1/2009	Levine	2010/0325655	A1	12/2010	Perez
2009/0044128	A1	2/2009	Baumgarten et al.	2010/0332401	A1	12/2010	Prahlad et al.
2009/0049131	A1	2/2009	Lyle et al.	2010/0332962	A1	12/2010	Hammer et al.
2009/0125469	A1	5/2009	McDonald et al.	2010/0333116	A1	12/2010	Prahlad et al.
2009/0132651	A1	5/2009	Roger et al.	2011/0001763	A1	1/2011	Murakami
2009/0138808	A1	5/2009	Moromisato et al.	2011/0016409	A1	1/2011	Grosz et al.
2009/0150417	A1	6/2009	Ghods et al.	2011/0022559	A1	1/2011	Andersen et al.
2009/0150627	A1	6/2009	Benhase et al.	2011/0022812	A1	1/2011	van der Linden et al.
2009/0158142	A1	6/2009	Arthursson et al.	2011/0029883	A1	2/2011	Lussier et al.
2009/0164438	A1	6/2009	Delacruz	2011/0040812	A1	2/2011	Phillips
2009/0171983	A1	7/2009	Samji et al.	2011/0041083	A1	2/2011	Gabai et al.
2009/0193107	A1	7/2009	Srinivasan et al.	2011/0047413	A1	2/2011	McGill et al.
2009/0193345	A1	7/2009	Wensley et al.	2011/0047484	A1	2/2011	Mount et al.
2009/0198772	A1	8/2009	Kim et al.	2011/0052155	A1	3/2011	Desmarais et al.
2009/0210459	A1	8/2009	Nair et al.	2011/0054968	A1	3/2011	Galaviz
2009/0214115	A1	8/2009	Kimura et al.	2011/0055299	A1	3/2011	Phillips
2009/0235167	A1	9/2009	Boyer et al.	2011/0055721	A1	3/2011	Jain et al.
2009/0235181	A1	9/2009	Saliba et al.	2011/0061045	A1	3/2011	Phillips
2009/0235189	A1	9/2009	Aybes et al.	2011/0061046	A1	3/2011	Phillips
				2011/0065082	A1	3/2011	Gal et al.
				2011/0066951	A1	3/2011	Ward-Karet et al.
				2011/0083167	A1	4/2011	Carpenter et al.
				2011/0093567	A1	4/2011	Jeon et al.

(56)

References Cited

U.S. PATENT DOCUMENTS

2011/0099006	A1	4/2011	Sundaraman et al.
2011/0113320	A1	5/2011	Neff et al.
2011/0119313	A1	5/2011	Sung et al.
2011/0137991	A1	6/2011	Russell
2011/0142410	A1	6/2011	Ishii
2011/0145744	A1	6/2011	Haynes et al.
2011/0161289	A1	6/2011	Pei et al.
2011/0167125	A1	7/2011	Achlioptas
2011/0167353	A1	7/2011	Grosz et al.
2011/0167435	A1	7/2011	Fang
2011/0185292	A1	7/2011	Chawla et al.
2011/0202424	A1	8/2011	Chun et al.
2011/0208958	A1	8/2011	Stuedi et al.
2011/0209064	A1	8/2011	Jorgensen et al.
2011/0213765	A1	9/2011	Cui et al.
2011/0219419	A1	9/2011	Reisman
2011/0225417	A1	9/2011	Maharajh et al.
2011/0238458	A1	9/2011	Purcell et al.
2011/0238621	A1	9/2011	Agrawal
2011/0238759	A1	9/2011	Spataro et al.
2011/0239135	A1	9/2011	Spataro et al.
2011/0246294	A1	10/2011	Robb et al.
2011/0246950	A1	10/2011	Luna et al.
2011/0252071	A1	10/2011	Cidon
2011/0252320	A1	10/2011	Arrasvuori et al.
2011/0258461	A1	10/2011	Bates
2011/0258561	A1	10/2011	Ladouceur et al.
2011/0282710	A1	11/2011	Akkiraju et al.
2011/0289433	A1	11/2011	Whalin et al.
2011/0296022	A1	12/2011	Ferris et al.
2011/0313803	A1	12/2011	Friend et al.
2011/0320197	A1	12/2011	Conejero et al.
2012/0036370	A1	2/2012	Lim et al.
2012/0064879	A1	3/2012	Panei
2012/0072436	A1	3/2012	Pierre et al.
2012/0079095	A1	3/2012	Evans et al.
2012/0089659	A1	4/2012	Halevi et al.
2012/0110005	A1	5/2012	Kuo et al.
2012/0110436	A1	5/2012	Adler, III et al.
2012/0117626	A1	5/2012	Yates et al.
2012/0124306	A1	5/2012	Abercrombie et al.
2012/0124547	A1	5/2012	Halbedel
2012/0130900	A1	5/2012	Tang et al.
2012/0134491	A1	5/2012	Liu
2012/0136936	A1	5/2012	Quintuna
2012/0144283	A1	6/2012	Hill et al.
2012/0150888	A1	6/2012	Hyatt et al.
2012/0151551	A1	6/2012	Readshaw et al.
2012/0158908	A1	6/2012	Luna et al.
2012/0159178	A1	6/2012	Lin et al.
2012/0159310	A1	6/2012	Chang et al.
2012/0173625	A1	7/2012	Berger
2012/0179981	A1	7/2012	Whalin et al.
2012/0185355	A1	7/2012	Kilroy
2012/0185913	A1	7/2012	Martinez et al.
2012/0192055	A1	7/2012	Antebi et al.
2012/0192086	A1	7/2012	Ghods et al.
2012/0203908	A1	8/2012	Beaty et al.
2012/0204032	A1	8/2012	Wilkins et al.
2012/0214444	A1	8/2012	McBride et al.
2012/0218885	A1	8/2012	Abel et al.
2012/0221789	A1	8/2012	Felter
2012/0226767	A1	9/2012	Luna et al.
2012/0233155	A1	9/2012	Gallmeier et al.
2012/0233205	A1	9/2012	McDermott
2012/0240061	A1	9/2012	Hillenius et al.
2012/0240183	A1	9/2012	Sinha
2012/0257249	A1	10/2012	Natarajan
2012/0263166	A1	10/2012	Cho et al.
2012/0266203	A1	10/2012	Elhadad et al.
2012/0284638	A1	11/2012	Cutler et al.
2012/0284664	A1	11/2012	Zhao
2012/0291011	A1	11/2012	Quine
2012/0309540	A1	12/2012	Holme et al.
2012/0311157	A1	12/2012	Erickson et al.
2012/0317239	A1	12/2012	Mulder et al.
2012/0317487	A1	12/2012	Lieb et al.
2012/0328259	A1	12/2012	Seibert, Jr. et al.
2012/0331177	A1	12/2012	Jensen
2012/0331441	A1	12/2012	Adamson
2013/0007245	A1	1/2013	Malik et al.
2013/0007471	A1	1/2013	Grab et al.
2013/0007894	A1	1/2013	Dang et al.
2013/0013560	A1	1/2013	Goldberg et al.
2013/0014023	A1	1/2013	Lee et al.
2013/0042106	A1	2/2013	Persaud et al.
2013/0055127	A1	2/2013	Saito et al.
2013/0067232	A1	3/2013	Cheung et al.
2013/0073403	A1	3/2013	Tuchman et al.
2013/0080919	A1	3/2013	Kiang et al.
2013/0117337	A1	5/2013	Dunham
2013/0124638	A1	5/2013	Barreto et al.
2013/0138608	A1	5/2013	Smith
2013/0138615	A1	5/2013	Gupta et al.
2013/0159411	A1	6/2013	Bowen
2013/0163289	A1	6/2013	Kim et al.
2013/0167253	A1	6/2013	Seleznnev et al.
2013/0185558	A1	7/2013	Seibert et al.
2013/0191339	A1	7/2013	Haden et al.
2013/0198600	A1	8/2013	Lockhart et al.
2013/0212486	A1	8/2013	Joshi et al.
2013/0239049	A1	9/2013	Perrodin et al.
2013/0246932	A1	9/2013	Zaveri et al.
2013/0262210	A1	10/2013	Savage et al.
2013/0262862	A1	10/2013	Hartley
2013/0268480	A1	10/2013	Dorman
2013/0268491	A1	10/2013	Chung et al.
2013/0275398	A1	10/2013	Dorman et al.
2013/0275429	A1	10/2013	York et al.
2013/0275509	A1	10/2013	Micucci et al.
2013/0305039	A1	11/2013	Gauda
2014/0007205	A1	1/2014	Oikonomou
2014/0013112	A1	1/2014	Cidon et al.
2014/0019497	A1	1/2014	Cidon et al.
2014/0019498	A1	1/2014	Cidon et al.
2014/0032616	A1	1/2014	Nack
2014/0033277	A1	1/2014	Xiao et al.
2014/0033291	A1	1/2014	Liu
2014/0052939	A1	2/2014	Tseng et al.
2014/0068589	A1	3/2014	Barak
2014/0156373	A1	6/2014	Roberts et al.
2014/0172595	A1	6/2014	Beddow et al.

FOREIGN PATENT DOCUMENTS

CN	102264063	A	11/2011
EP	0921661	A2	6/1999
EP	1349088		10/2003
EP	1528746	A2	5/2005
EP	2372574	A1	10/2011
EP	2610776	A2	7/2013
GB	2453924	A	4/2009
GB	2471282	A	12/2010
JP	09-101937		4/1997
JP	11-025059		1/1999
JP	2003273912	A	9/2003
JP	2004310272	A	11/2004
JP	09-269925		10/2007
JP	2008250944	A	10/2008
KR	20020017444	A	3/2002
KR	20040028036	A	4/2004
KR	20050017674	A	2/2005
KR	20060070306	A	6/2006
KR	20060114871	A	11/2006
KR	20070043353	A	4/2007
KR	20070100477	A	10/2007
KR	20100118836	A	11/2010
KR	20110074096	A	6/2011
KR	20110076831	A	7/2011
WO	WO-0007104	A1	2/2000
WO	WO-0219128	A1	3/2002
WO	WO-2004097681	A1	11/2004
WO	WO-2006028850	A2	3/2006
WO	WO-2007024438	A1	3/2007

(56)

References Cited

FOREIGN PATENT DOCUMENTS

WO WO-2007035637 A2 3/2007
 WO WO-2007113573 A2 10/2007
 WO WO-2008011142 A2 1/2008
 WO WO-2008076520 A2 6/2008
 WO WO-2011109416 A2 9/2011
 WO WO-2012167272 A1 12/2012
 WO WO-2013009328 A2 1/2013
 WO WO-2013013217 A1 1/2013
 WO WO-2013041763 A1 3/2013
 WO WO-2013166520 A1 11/2013

OTHER PUBLICATIONS

"How-to Geek, How to Sync Specific Folders With Dropbox," downloaded from the internet <http://www.howtogeek.com>, Apr. 23, 2013, 5 pages.

"Microsoft Office SharePoint 2007 User Guide," Feb. 16, 2010, pp. 1-48.

"Understanding Metadata," National Information Standards Organization, NISO Press, 2004, 20 pages.

Cisco, "FTP Load Balancing on ACE in Routed Mode Configuration Example," DocWiki, Jun. 2011, 7 pages.

Conner, "Google Apps: The Missing Manual," published by O'Reilly Media, May 27, 2008, 24 pages.

Exam Report for EP13158415.3, Applicant: Box, Inc. Mailed Jun. 4, 2013, 8 pages.

Exam Report for GB1300188.8, Applicant: Box, Inc. Mailed May 31, 2013, 8 pages.

Exam Report for GB1306011.6, Applicant: Box, Inc. Mailed Apr. 18, 2013, 8 pages.

Exam Report for GB1310666.1, Applicant: Box, Inc. Mailed Aug. 30, 2013, 10 pages.

Exam Report for GB1313559.5, Applicant: Box, Inc., Mailed Aug. 22, 2013, 19 pages.

Google Docs, http://web.archive.org/web/20100413105758/http://en.wikipedia.org/wiki/Google_docs, Apr. 13, 2010, 6 pages.

International Search Report and Written Opinion for PCT/US2008/012973 dated Apr. 30, 2009, pp. 1-11.

International Search Report and Written Opinion for PCT/US2011/039126 mailed on Oct. 6, 2011, pp. 1-13.

International Search Report and Written Opinion for PCT/US2011/041308 Mailed Jul. 2, 2012, pp. 1-16.

International Search Report and Written Opinion for PCT/US2011/047530, Applicant: Box, Inc., Mailed Mar. 22, 2013, pp. 1-10.

International Search Report and Written Opinion for PCT/US2011/056472 mailed on Jun. 22, 2012, pp. 1-12.

International Search Report and Written Opinion for PCT/US2011/057938, Applicant: Box, Inc., Mailed Mar. 29, 2013, 10 pages.

International Search Report and Written Opinion for PCT/US2011/060875 Mailed Oct. 30, 2012, pp. 1-10.

International Search Report and Written Opinion for PCT/US2012/056955, Applicant: Box, Inc., Mailed Mar. 27, 2013, pp. 1-11.

International Search Report and Written Opinion for PCT/US2012/063041, Applicant: Box, Inc., Mailed Mar. 29, 2013, 12 pages.

International Search Report and Written Opinion for PCT/US2012/065617, Applicant: Box, Inc., Mailed Mar. 29, 2013, 9 pages.

International Search Report and Written Opinion for PCT/US2012/067126, Applicant: Box, Inc., Mailed Mar. 29, 2013, 10 pages.

International Search Report and Written Opinion for PCT/US2012/070366, Applicant: Box, Inc., Mailed Apr. 24, 2013, 10 pages.

International Search Report and Written Opinion for PCT/US2013/020267, Applicant: Box, Inc., Mailed May 7, 2013, 10 pages.

International Search Report and Written Opinion for PCT/US2013/023889, Applicant: Box, Inc., Mailed Jun. 24, 2013, 13 pages.

International Search Report and Written Opinion for PCT/US2013/029520, Applicant: Box, Inc., Mailed Jun. 26, 2013, 10 pages.

International Search Report and Written Opinion for PCT/US2013/034662, Applicant: Box, Inc., Mailed May 31, 2013, 10 pages.

International Search Report and Written Opinion for PCT/US2013/035404, Applicant: Box, Inc., Mailed Jun. 26, 2013, 13 pages.

International Search Report and Written Opinion for PCT/US2013/039782, Applicant: Box, Inc., Mailed Aug. 28, 2013, 15 pages.

Internet Forums, http://web.archive.org/web/20100528195550/http://en.wikipedia.org/wiki/Internet_forums, Wikipedia, May 30, 2010, pp. 1-20.

Langfeld L. et al., "Microsoft SharePoint 2003 Unleashed," Chapters 11 and 15, Jun. 2004, pp. 403-404, 557-561, 578-581.

Lars, "35 Very Useful Online Tools for Improving your project Management and Team Collaboration," Apr. 31, 2010, tripwiremagazine.com, pp. 1-32.

Palmer, "Load Balancing FTP Servers," BlogNav, Oct. 2008, 2 pages.

Parr, "Google Docs Improves Commenting, Adds E-mail Notifications," Apr. 16, 2011, mashable.com, pp. 1-6.

Partial International Search Report for PCT/US2011/041308 dated Feb. 27, 2012, pp. 1-2.

Supplementary European Search Report European Application No. EP 08 85 8563 dated Jun. 20, 2011 pp. 1-5.

Wayback, "Wayback machine," Wayback, Jun. 1, 2011, 1 page.

Wiki, <http://web.archive.org/web/20100213004936/http://en.wikipedia.org/wiki/Wiki>, Feb. 13, 2010, pp. 1-16.

Yahoo! Groups, http://web.archive.org/web/20090320101529/http://en.wikipedia.org/wiki/Yahoo!_Groups, Wikipedia, Mar. 20, 2009, pp. 1-6.

"PaperPort Professional 14," PC Mag. Com review, published Feb. 2012, Ziff Davis, Inc., 8 pages.

"PaperPort," Wikipedia article (old revision), published May 19, 2012, Wikipedia Foundation, 2 pages.

"Quickoffice Enhances Android Mobile office Application for Improved Productivity on latest Smartphone and Table Devices," QuickOffice Press Release, Nov. 21, 2011, QuickOffice Inc., 2 pages.

"QuickOffice," Wikipedia Article (old revision), published May 9, 2012, Wikipedia Foundation, 2 pages.

Exam Report for EP13168784.0, Applicant: Box, Inc. Mailed Nov. 21, 2013, 7 pages.

Exam Report for EP13185269.1, Applicant: Box, Inc. Mailed Jan. 28, 2013, 7 pages.

Exam Report for GB1309209.3, Applicant: Box, Inc. Mailed Oct. 30, 2013, 11 pages.

Exam Report for GB1311417.8, Applicant: Box, Inc. Mailed Dec. 23, 2013, 5 pages.

Exam Report for GB1312095.1, Applicant: Box, Inc. Mailed Dec. 12, 2013, 7 pages.

Exam Report for GB1312874.9, Applicant: Box, Inc. Mailed Dec. 20, 2013, 11 pages.

Exam Report for GB1316532.9, Applicant: Box, Inc. Mailed Oct. 31, 2013, 10 pages.

Exam Report for GB1316533.7, Applicant: Box, Inc. Mailed Oct. 8, 2013, 9 pages.

Exam Report for GB1316971.9, Applicant: Box, Inc. Mailed Nov. 26, 2013, 10 pages.

Exam Report for GB1317600.3, Applicant: Box, Inc. Mailed Nov. 21, 2013, 8 pages.

Exam Report for GB1320902.8, Applicant: Box, Inc. Mailed Dec. 20, 2013, 6 pages.

Gedymin, "Cloud computing with an emphasis on Google App Engine," Master Final Project, Sep. 2011, 146 pages.

International Search Report and Written Opinion for PCT/US2013/034765, Applicant: Box, Inc., Mailed Jan. 20, 2014, 15 pages.

Patent Court Document of Approved Judgment for GB0602349.3 and GB0623571.7; Mar. 3, 2009, 17 pages.

Exam Report for GB1410569.6 Applicant: Box, Inc. Mailed Jul. 11, 2014, 9 pages.

Extended Search Report for EP131832800, Applicant: Box, Inc. Mailed Aug. 25, 2014, 7 pages.

Extended Search Report for EP141509422, Applicant: Box, Inc. Mailed Aug. 26, 2014, 12 pages.

Search Report for EP 13189144.2 Applicant: Box, Inc. Mailed Sep. 1, 2014, 9 pages.

Exam Report for GB1312874.9 Applicant: Box, Inc. Mailed Sep. 26, 2014, 2 pages.

Exam Report for GB1415126.0 Applicant: Box, Inc. Mailed Oct. 2, 2014, 8 pages.

(56)

References Cited

OTHER PUBLICATIONS

Exam Report for GB1415314.2 Applicant: Box, Inc. Mailed Oct. 7, 2014, 6 pages.

Exam Report for GB1309209.3 Applicant: Box, Inc. Mailed Oct. 7, 2014, 3 pages.

Exam Report for GB1315232.7 Applicant: Box, Inc. Mailed Oct. 9, 2014, 5 pages.

Exam Report for GB1318789.3 Applicant: Box, Inc. Mailed Oct. 30, 2014, 6 pages.

Microsoft Windows XP Professional Product Documentation: How Inheritance Affects File and Folder Permissions, Apr. 11, 2014, 2 pages.

Exam Report for GB1317393.5 Applicant: Box, Inc. Mailed Nov. 7, 2014, 6 pages.

Exam Report for GB1311417.8 Applicant: Box, Inc. Mailed Nov. 7, 2014, 2 pages.

Exam Report for GB1311421.0 Applicant: Box, Inc. Mailed Nov. 7, 2014, 4 pages.

Exam Report for GB1316682.2 Applicant: Box, Inc. Mailed Nov. 19, 2014, 6 pages.

Exam Report for GB1312095.1 Applicant: Box, Inc. Mailed Nov. 19, 2014, 5 pages.

Exam Report for GB1313559.5 Applicant: Box, Inc. Mailed Nov. 4, 2014, 2 pages.

User's Guide for SMART Board Software for Windows, published Dec. 2004, 90 pages.

Zambonini et al., "Automated Measuring of Interaction with User Interfaces," Published as WOWO2007113573 Oct. 2007, 19 pages.

Exam Report for GB1314771.5, Applicant: Box, Inc. Mailed Feb. 17, 2014, 7 pages.

Exam Report for GB1308842.2, Applicant: Box, Inc. Mailed Mar. 10, 2014, 4 pages.

Burns, "Developing Secure Mobile Applications for Android," Oct. 2008, Version 1.0, 1-28 pages.

Search Report for EP 11729851.3, Applicant: Box, Inc. Mailed Feb. 7, 2014, 9 pages.

Comes, "MediaXchange User's Manual," Version 1.15.15, Feb. 1, 2009, pp. 1-90.

"Average Conversion Time for a D60 RAW file?" <http://www.dpreview.com>, Jul. 22, 2002, 4 pages.

Exam Report for GB1312264.3, Applicant: Box, Inc. Mailed Mar. 24, 2014, 7 pages.

Search Report for EP14153783.7, Applicant: Box, Inc. Mailed Mar. 24, 2014, 7 pages.

John et al., "Always Sync Support Forums—View topic—Allway sync funny behavior," Allway Sync Support Forum at <http://sync-center.com>, Mar. 28, 2011, XP055109680, 2 pages.

Search Report for EP14151588.2, Applicant: Box, Inc. Mailed Apr. 15, 2014, 12 pages.

Search Report for EP13187217.8, Applicant: Box, Inc. Mailed Apr. 15, 2014, 12 pages.

Rao, "Box Acquires Crocodoc To Add HTML5 Document Converter And Sleek Content Viewing Experience To Cloud Storage Platform," retrieved from the internet, <http://techcrunch.com>, May 9, 2013, 8 pages.

Walker, "PDF.js project meeting notes," retrieved from the internet, <http://groups.google.com>, May 15, 2014, 1 page.

Sommerer, "Presentable Document Format: Improved On-demand PDF to HTML Conversion," retrieved from the internet, <http://research.microsoft.com>, 8 pages.

"Tulsa TechFest 2012—Agenda," retrieved from the website, <http://web.archive.org>, Oct. 2, 2012, 2 pages.

Delendik, "Evolving with Web Standards—The Story of PDF.JS," retrieved from the internet, <http://people.mozilla.org>, Oct. 12, 2012, 36 pages.

Delendik, "My PDF.js talk slides from Tulsa TechFest," retrieved from the internet, <http://twitter.com>, Oct. 12, 2012, 2 pages.

Cohen, "Debating the Definition of Cloud Computing Platforms," retrieved from the internet, <http://forbes.com>, Feb. 3, 2014, 7 pages.

Partial Search Report for EP131832800, Applicant: Box, Inc. Mailed May 8, 2014, 5 pages.

Search Report for EP141509422, Applicant: Box, Inc. Mailed May 8, 2014, 7 pages.

Tulloch et al., "Windows Vista Resource Kit," Apr. 8, 2007, Microsoft Press, XP055113067, 6 pages.

"Revolving sync conflicts; frequently asked questions," Microsoft Tech Support, Jul. 16, 2012, retrieved from the Internet: <http://web.archive.org/web>, 2 pages.

Pyle et al., "How to enable Event logging for Offline Files (Client Side Caching) in Windows Vista," Feb. 18, 2009, retrieved from the internet: <http://blogs.technet.com>, 3 pages.

Duffy, "The Best File-Syncing Services," pcmag.com, retrieved from the internet: <http://www.pcmag.com>, Sep. 28, 2012, 7 pages.

"Troubleshoot sync problems," Microsoft Tech Support: May 2, 2012, retrieved from the internet, <http://web.archive.org/web>, 3 pages.

Exam Report for GB1318792.7, Applicant: Box, Inc. Mailed May 22, 2014, 2 pages.

Exam Report for EP13177108.1, Applicant: Box, Inc. Mailed May 26, 2014, 6 pages.

* cited by examiner

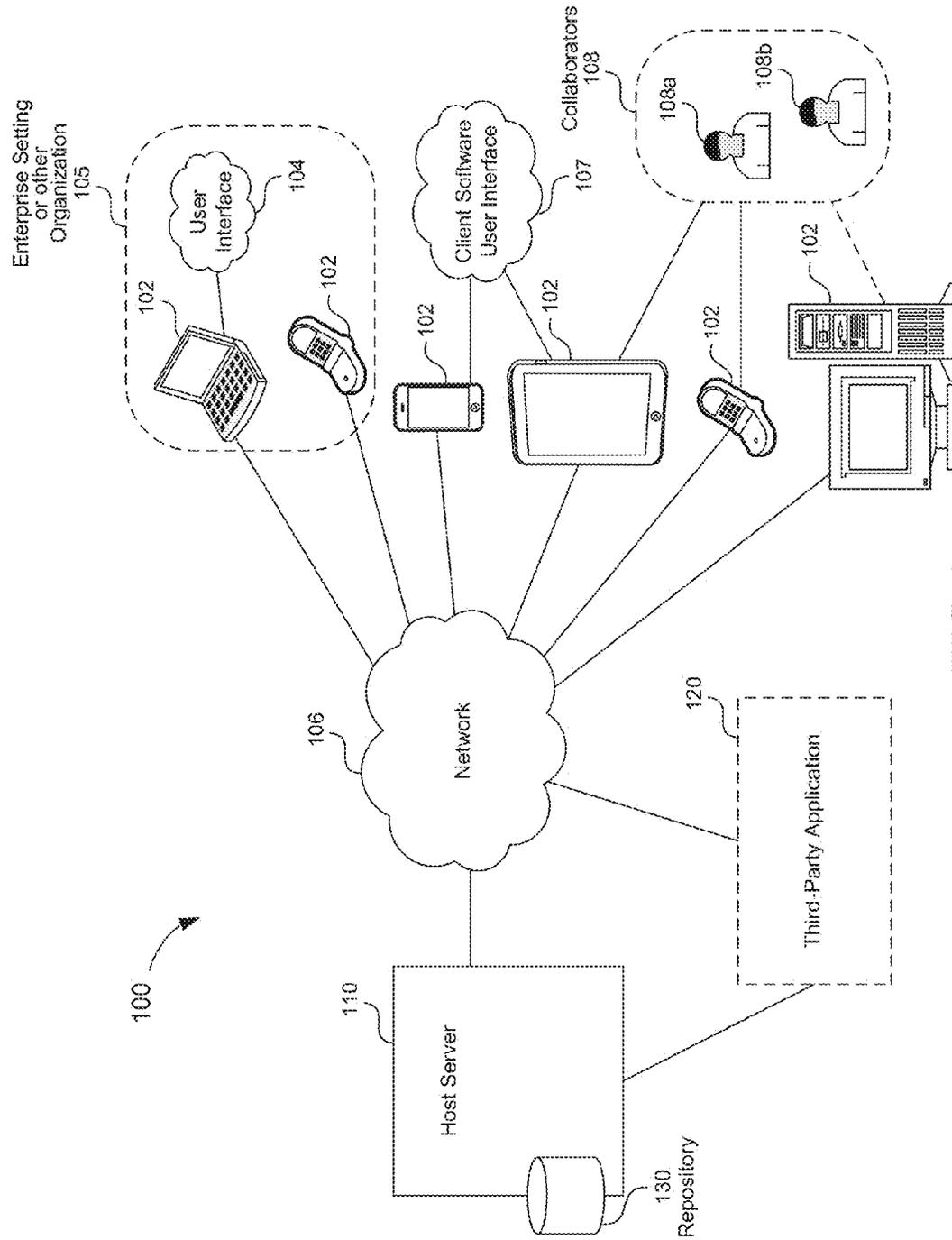


FIG. 1

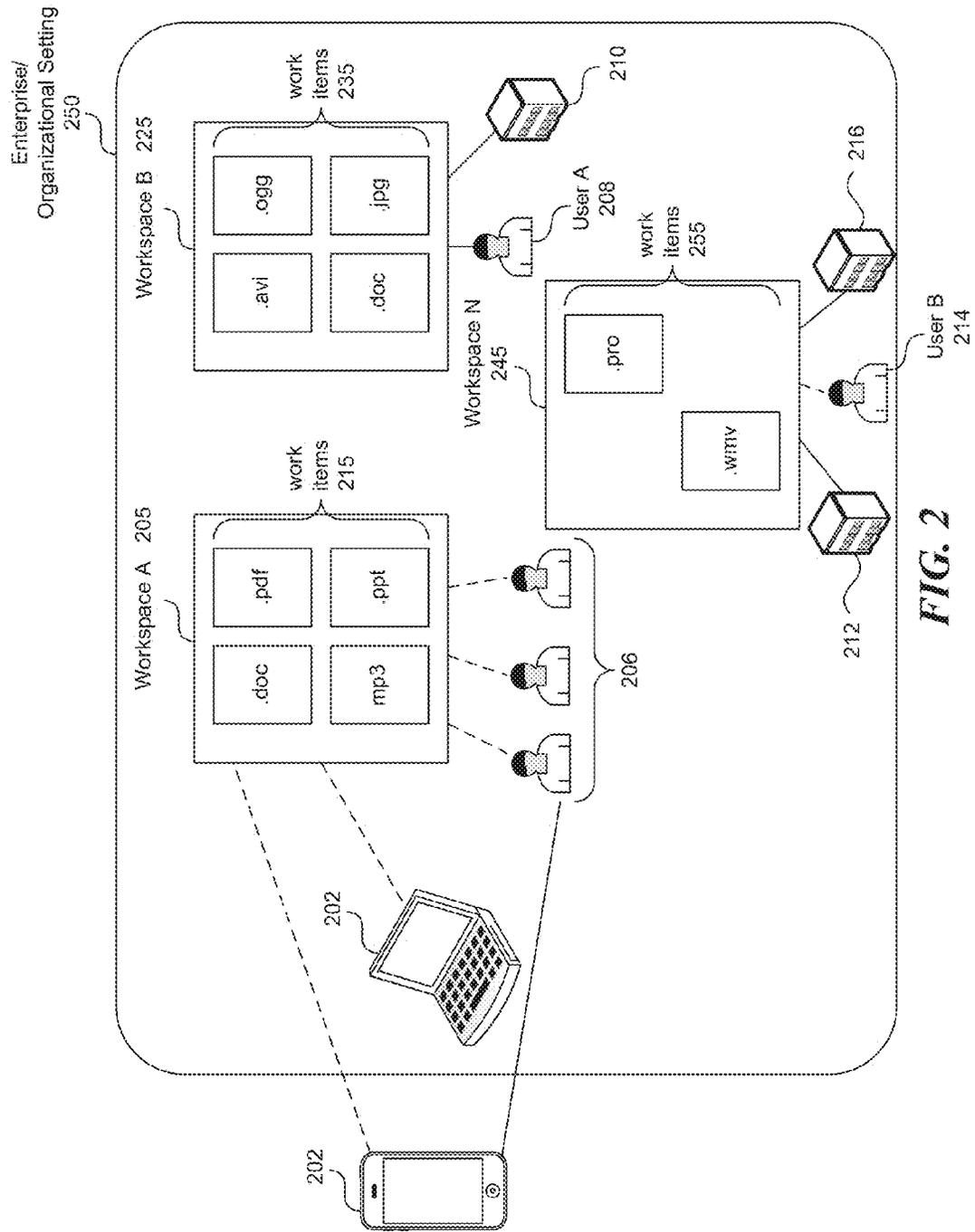


FIG. 2

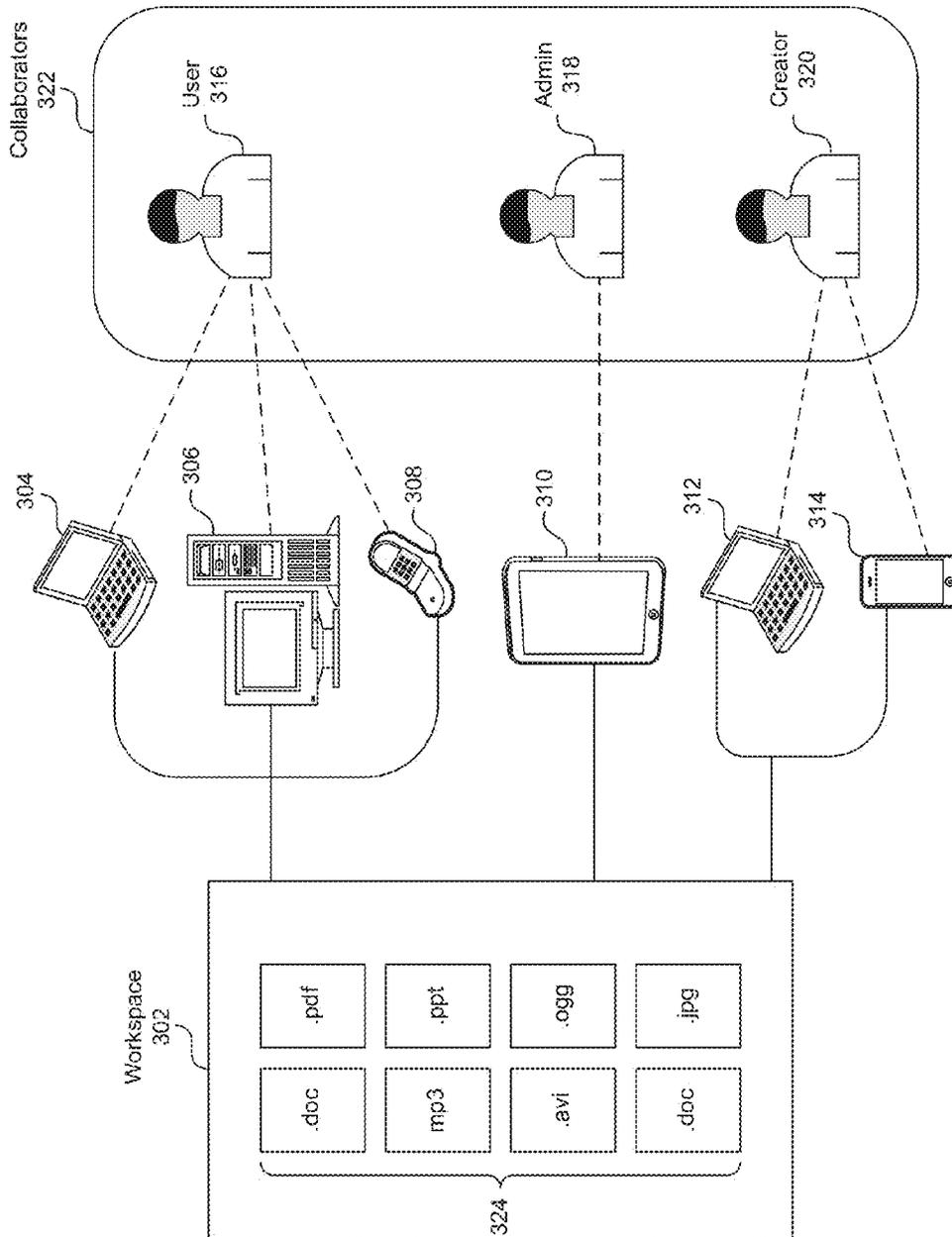


FIG. 3

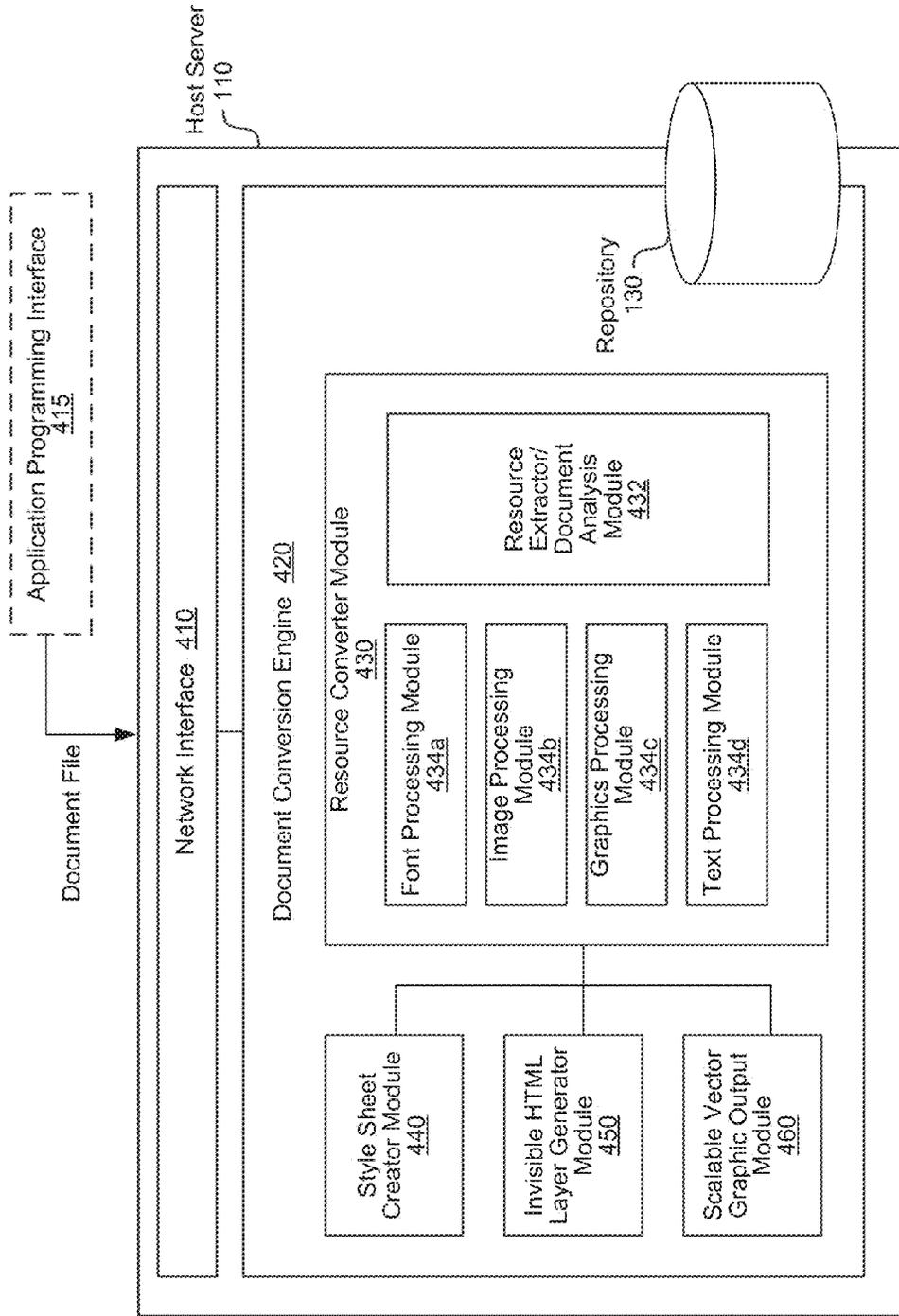


FIG. 4

500

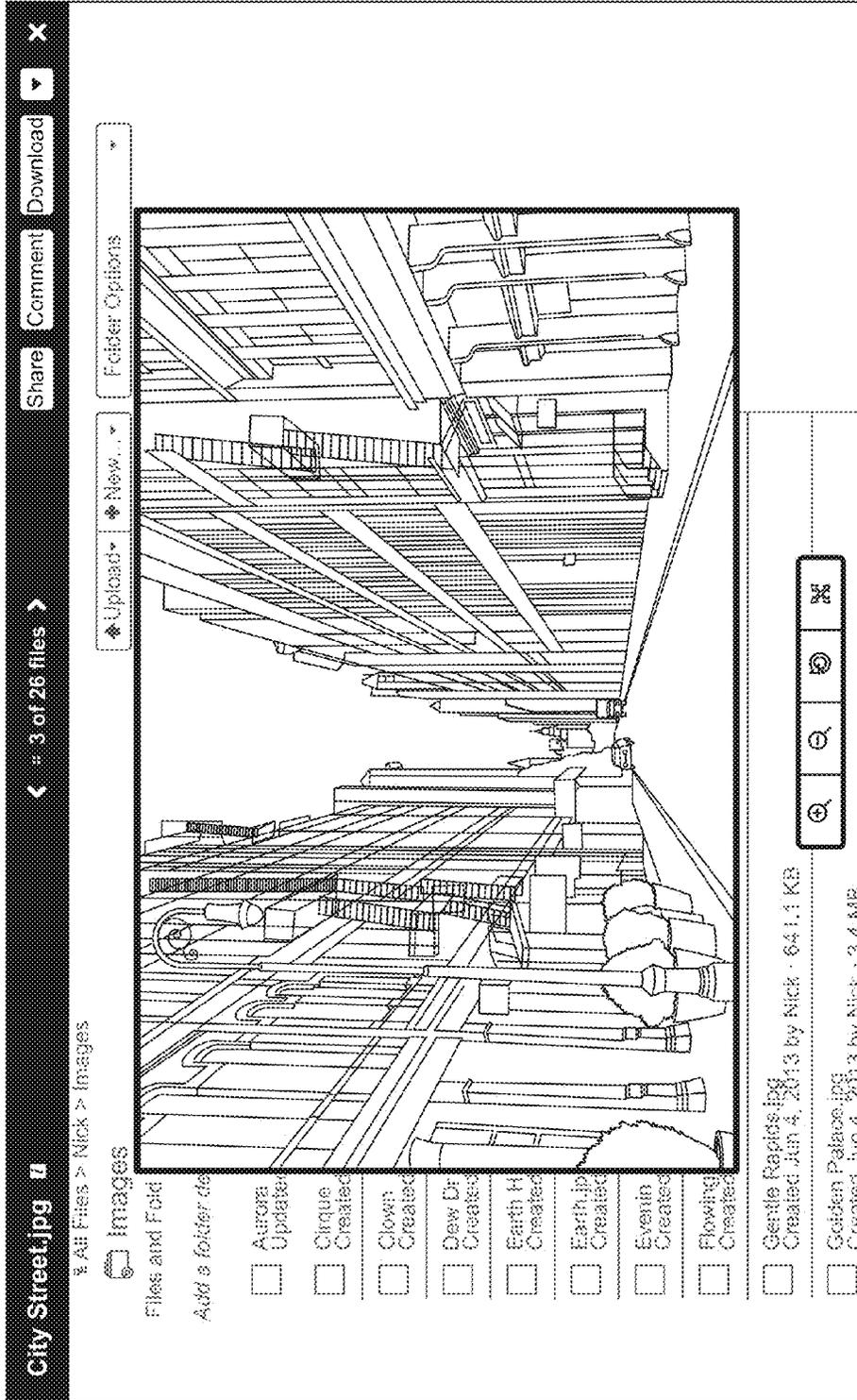


FIG. 5A

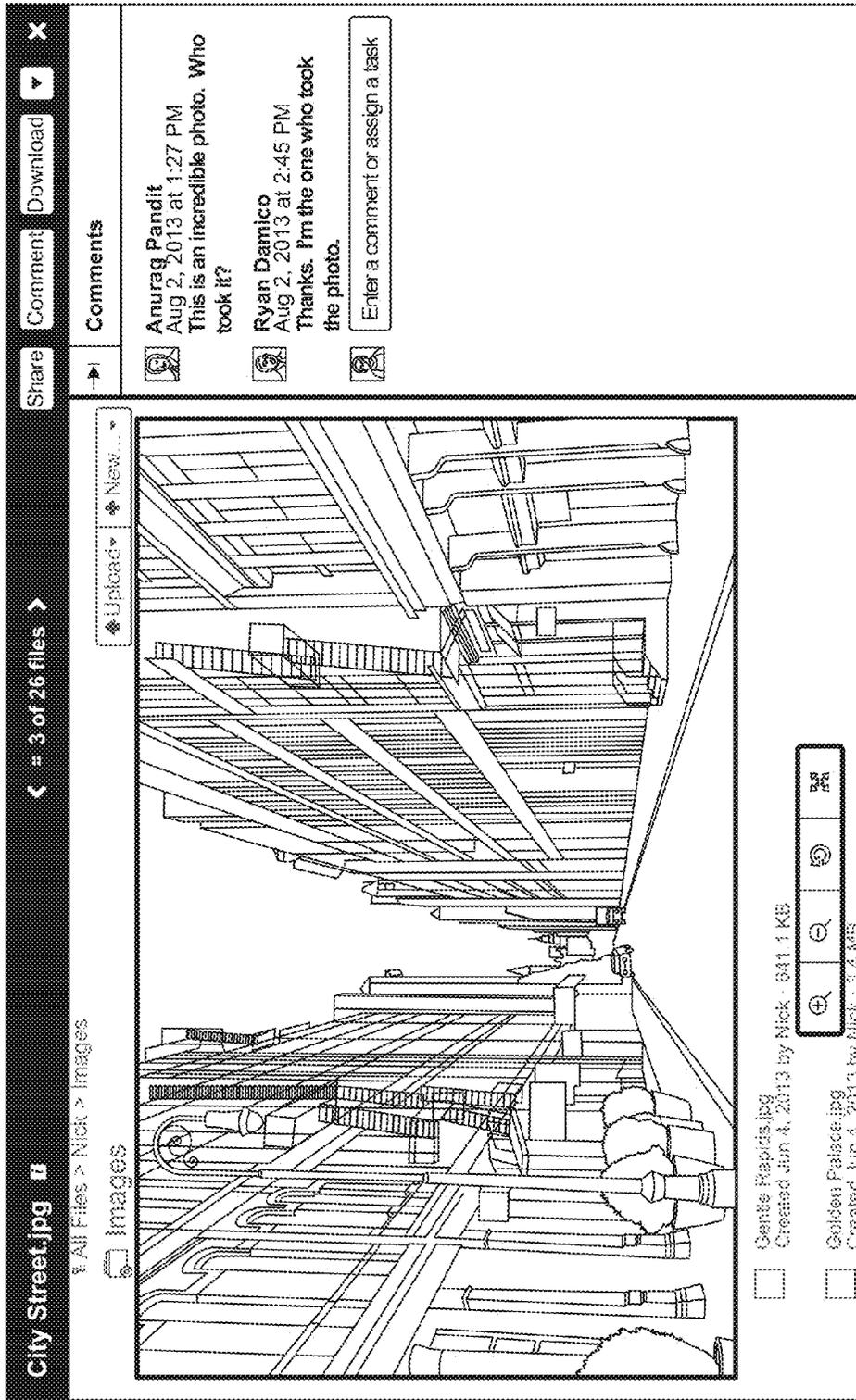


FIG. 5C

520

600

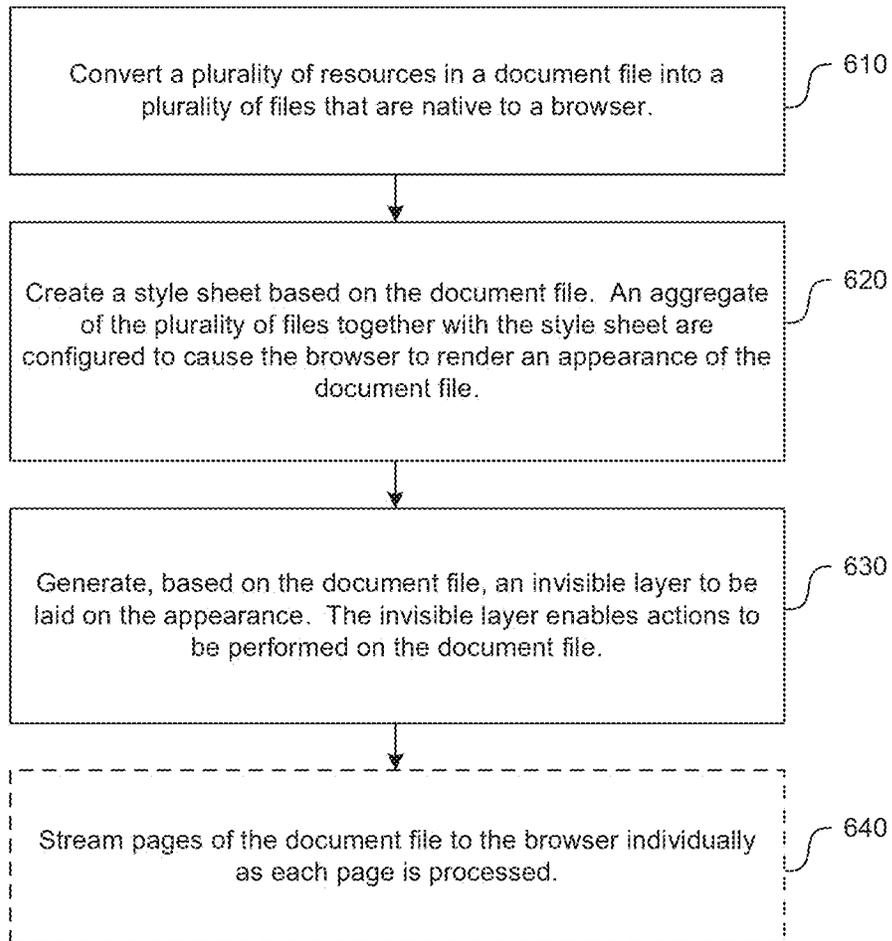


FIG. 6

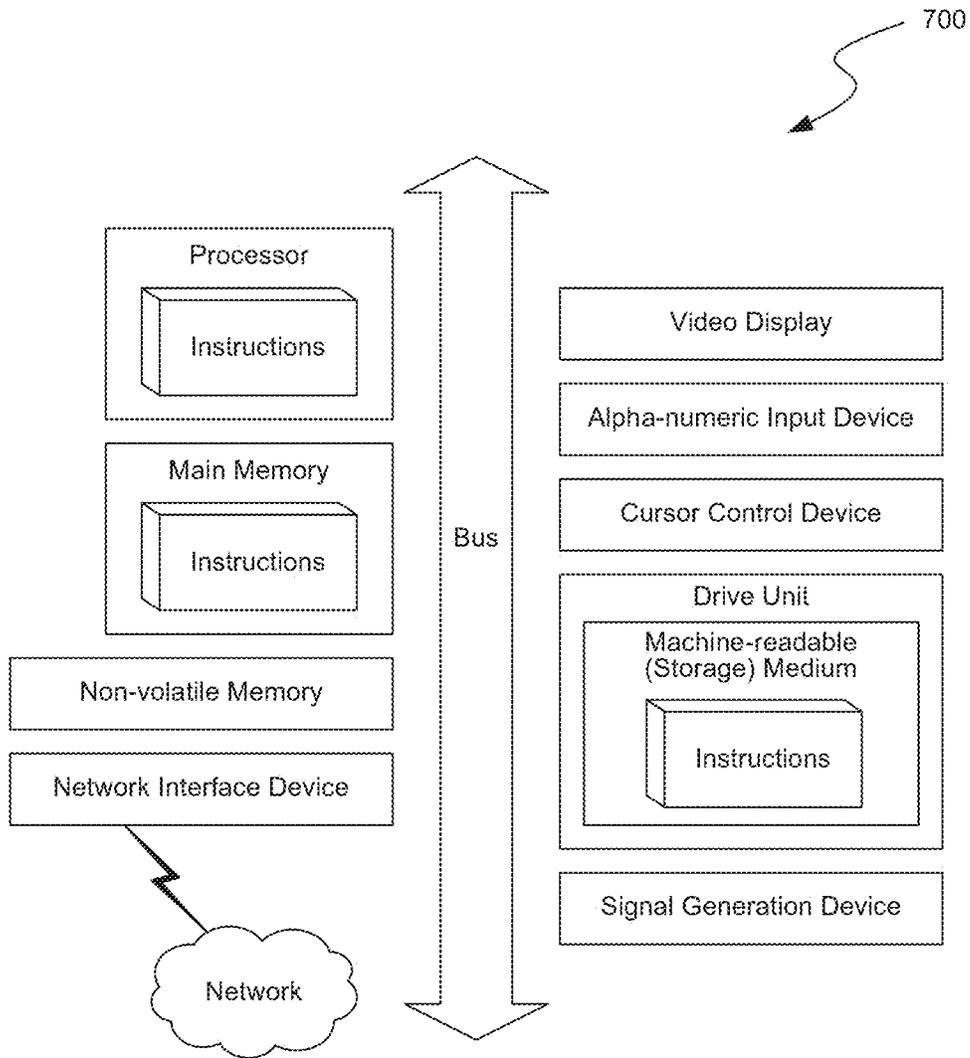


FIG. 7

1

**SYSTEM AND METHOD FOR RENDERING
DOCUMENT IN WEB BROWSER OR MOBILE
DEVICE REGARDLESS OF THIRD-PARTY
PLUG-IN SOFTWARE**

COPYRIGHT NOTICE

A portion of the disclosure of this patent document contains material that is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the United States Patent and Trademark Office patent files or records, but otherwise reserves all copyright rights whatsoever. The following notice applies to the software and data as described below and in the drawings that form a part of this document:

Copyright 2013, Box, Inc., All Rights Reserved.

BACKGROUND

With the advancements in digital technologies, data proliferation and the ever increasing mobility of user platforms have created enormous amounts of information traffic over mobile and computer networks. This is particularly relevant to the increase of electronic and digital content being used and shared over the network in social settings and working environments as compared to traditional stand-alone personal computers and mobile devices. As a result, content is shared across multiple devices among multiple users.

However, among others, both the wide variety in file types and device types place a barrier to easy access of the shared content. Before accessing a file (e.g., in a proprietary document format) that is foreign to a user's system, the user typically must install a software (e.g., an executable or a plug-in) onto the user's system. Additionally, the user may be required to complete a time-consuming setup process which further impedes the access.

BRIEF DESCRIPTION OF DRAWINGS

The present embodiments are illustrated by way of example and are not intended to be limited by the figures of the accompanying drawings. In the drawings:

FIG. 1 depicts an example diagram of a system having a host server of a cloud service, collaboration and/or cloud storage accounts with capabilities that render documents in a browser or mobile device regardless of whether a plug-in therefor is installed;

FIG. 2 depicts an example diagram of a web-based or online collaboration platform deployed in an enterprise or other organizational setting for organizing work items and workspaces;

FIG. 3 depicts an example diagram of a workspace in an online or web-based collaboration environment accessible by multiple collaborators through various devices;

FIG. 4 depicts a block diagram illustrating an example of components in a host server with capabilities that render documents in a browser or mobile device regardless of whether a plug-in therefor is installed;

FIG. 5A-5C respectively depict screenshots showing example user interfaces embodying one or more techniques disclosed herein for rendering documents in a browser or mobile device regardless of whether a plug-in therefor is installed;

FIG. 6 depicts a flowchart illustrating an example process for rendering documents in a browser or mobile device regardless of whether a plug-in therefor is installed; and

2

FIG. 7 depicts a diagrammatic representation of a machine in the example form of a computer system within which a set of instructions, for causing the machine to perform any one or more of the methodologies discussed herein, can be executed.

The same reference numbers and any acronyms identify elements or acts with the same or similar structure or functionality throughout the drawings and specification for ease of understanding and convenience.

DETAILED DESCRIPTION

The following description and drawings are illustrative and are not to be construed as limiting. Numerous specific details are described to provide a thorough understanding of the disclosure. However, in certain instances, well-known or conventional details are not described in order to avoid obscuring the description. References to one or an embodiment in the present disclosure can be, but not necessarily are, references to the same embodiment; and, such references mean at least one of the embodiments.

Reference in this specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the disclosure. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment, nor are separate or alternative embodiments mutually exclusive of other embodiments. Moreover, various features are described which can be exhibited by some embodiments and not by others. Similarly, various requirements are described which can be requirements for some embodiments but not other embodiments.

The terms used in this specification generally have their ordinary meanings in the art, within the context of the disclosure, and in the specific context where each term is used. Certain terms that are used to describe the disclosure are discussed below, or elsewhere in the specification, to provide additional guidance to the practitioner regarding the description of the disclosure. For convenience, certain terms can be highlighted, for example using italics and/or quotation marks. The use of highlighting has no influence on the scope and meaning of a term; the scope and meaning of a term is the same, in the same context, whether or not it is highlighted. It will be appreciated that same thing can be said in more than one way.

Consequently, alternative language and synonyms can be used for any one or more of the terms discussed herein, nor is any special significance to be placed upon whether or not a term is elaborated or discussed herein. Synonyms for certain terms are provided. A recital of one or more synonyms does not exclude the use of other synonyms. The use of examples anywhere in this specification including examples of any terms discussed herein is illustrative only, and is not intended to further limit the scope and meaning of the disclosure or of any exemplified term. Likewise, the disclosure is not limited to various embodiments given in this specification.

Without intent to limit the scope of the disclosure, examples of instruments, apparatus, methods and their related results according to the embodiments of the present disclosure are given below. Note that titles or subtitles can be used in the examples for convenience of a reader, which in no way should limit the scope of the disclosure. Unless otherwise defined, all technical and scientific terms used herein have the same meaning as commonly understood by one of ordinary skill in the art to which this disclosure pertains. In the case of conflict, the present document, including definitions will control.

Techniques are disclosed for render documents that are in proprietary formats in a browser or mobile device. In one embodiment, a method comprises converting a plurality of resources in a document file into a plurality of files that are native to a browser. The method further comprises creating a style sheet based on the document file, wherein an aggregate of the plurality of files together with the style sheet are configured to cause the browser to render an appearance of the document file. The method further comprises generating, based on the document file, an invisible layer to be laid on the appearance, wherein the invisible layer enables actions to be performed on the document file.

Among other advantages, embodiments disclosed herein provide the convenience of viewing and accessing documents regardless of whether a software or plug-in therefor is installed, thereby enabling easy access to document files (e.g., that are stored in the cloud-based platform or simply provided from an application for conversion), reducing or eliminating the time for additional download, installation and setup time of third-party software/plug-ins, which in turn enhances user experience and productivity.

FIG. 1 illustrates an example diagram of a system 100 having a host server 110 of a cloud service/platform, collaboration and/or cloud storage service with capabilities that render documents in a browser or mobile device regardless of whether a plug-in therefor is installed.

The client devices 102 can be any system and/or device, and/or any combination of devices/systems that is able to establish a connection, including wired, wireless, cellular connections with another device, a server and/or other systems such as host server 110 and/or a third-party application 120. Client devices 102 typically include a display and/or other output functionalities to present information and data exchanged between among the devices 102, the third-party application 120, and/or the host server 110.

For example, the client devices 102 can include mobile, hand held or portable devices or non-portable devices and can be any of, but not limited to, a server desktop, a desktop computer, a computer cluster, or portable devices including, a notebook, a laptop computer, a handheld computer, a palmtop computer, a mobile phone, a cell phone, a PDA, a smart phone (e.g., a BlackBerry device such as BlackBerry Z10/Q10, an iPhone, Nexus 4, etc.), a Treo, a handheld tablet (e.g. an iPad, iPad Mini, a Galaxy Note, Galaxy Note II, Xoom Tablet, Microsoft Surface, Blackberry PlayBook, Nexus 7, 10 etc.), a phablet (e.g., HTC Droid DNA, etc.), a tablet PC, a thin-client, a hand held console, a hand held gaming device or console (e.g., XBOX live, Nintendo DS, Sony PlayStation Portable, etc.), iOS powered watch, Google Glass, a Chromebook and/or any other portable, mobile, hand held devices, etc. running on any platform or any operating system (e.g., Mac-based OS (OS X, iOS, etc.), Windows-based OS (Windows Mobile, Windows 7, Windows 8, etc.), Android, Blackberry OS, Embedded Linux platforms, Palm OS, Symbian platform, Google Chrome OS, and the like. In one embodiment, the client devices 102, and host server 110 are coupled via a network 106. In some embodiments, the devices 102 and host server 110 can be directly connected to one another.

The input mechanism on client devices 102 can include touch screen keypad (including single touch, multi-touch, gesture sensing in 2D or 3D, etc.), a physical keypad, a mouse, a pointer, a track pad, motion detector (e.g., including 1-axis, 2-axis, 3-axis accelerometer, etc.), a light sensor, capacitance sensor, resistance sensor, temperature sensor, proximity sensor, a piezoelectric device, device orientation detector (e.g., electronic compass, tilt sensor, rotation sensor, gyroscope, accelerometer), or a combination of the above.

Signals received or detected indicating user activity at client devices 102 through one or more of the above input mechanism, or others, can be used by various users or collaborators (e.g., collaborators 108) for accessing, through network 106, a web-based collaboration environment or online collaboration platform (e.g., hosted by the host server 110). The collaboration environment or platform can have one or more collective settings 105 for an enterprise or an organization that the users belong, and can provide an user interface 104 (e.g., via a webpage application accessible by the web browsers of devices 102) for the users to access such platform under the settings 105. Additionally, a client software that is native to the cloud collaboration platform can be provided (e.g., through downloading from the host server 110 via the network 106) to run on the client devices 102 to provide cloud-based platform access functionalities. The users and/or collaborators can access the collaboration platform via a client software user interface 107, which can be provided by the execution of the client software on the devices 102.

The collaboration platform or environment hosts workspaces with work items that one or more users can access (e.g., view, edit, update, revise, comment, download, preview, tag, or otherwise manipulate, etc.). A work item can generally include any type of digital or electronic content that can be viewed or accessed via an electronic device (e.g., device 102). The digital content can include .pdf files, .doc, slides (e.g., Powerpoint slides), images, audio files, multimedia content, web pages, blogs, etc. A workspace can generally refer to any grouping of a set of digital content in the collaboration platform. The grouping can be created, identified, or specified by a user or through other means. This user can be a creator user or administrative user, for example.

In general, a workspace can be associated with a set of users or collaborators (e.g., collaborators 108) which have access to the content included therein. The levels of access (e.g., based on permissions or rules) of each user or collaborator to access the content in a given workspace can be the same or can vary among the users. Each user can have their own set of access rights to every piece of content in the workspace, or each user can be different access rights to different pieces of content. Access rights can be specified by a user associated with a workspace and/or a user who created/uploaded a particular piece of content to the workspace, or any other designated user or collaborator.

In general, the collaboration platform allows multiple users or collaborators to access or collaborate efforts on work items such each user can see, remotely, edits, revisions, comments, or annotations being made to specific work items through their own user devices. For example, a user can upload a document to a workspace for other users to access (e.g., for viewing, editing, commenting, signing-off, or otherwise manipulating). The user can login to the online platform and upload the document (or any other type of work item) to an existing workspace or to a new workspace. The document can be shared with existing users or collaborators in a workspace.

In general, network 106, over which the client devices 102 and the host server 110 communicate can be a cellular network, a telephonic network, an open network, such as the Internet, or a private network, such as an intranet and/or the extranet, or any combination or variation thereof. For example, the Internet can provide file transfer, remote log in, email, news, RSS, cloud-based services, instant messaging, visual voicemail, push mail, VoIP, and other services through any known or convenient protocol, such as, but is not limited

to the TCP/IP protocol, Open System Interconnections (OSI), FTP, UPnP, iSCSI, NSF, ISDN, PDH, RS-232, SDH, SONET, etc.

The network **106** can be any collection of distinct networks operating wholly or partially in conjunction to provide connectivity to the client devices **102** and the host server **110** and can appear as one or more networks to the serviced systems and devices. In one embodiment, communications to and from the client devices **102** can be achieved by, an open network, such as the Internet, or a private network, such as an intranet and/or the extranet. In one embodiment, communications can be achieved by a secure communications protocol, such as secure sockets layer (SSL), or transport layer security (TLS).

In addition, communications can be achieved via one or more networks, such as, but are not limited to, one or more of WiMax, a Local Area Network (LAN), Wireless Local Area Network (WLAN), a Personal area network (PAN), a Campus area network (CAN), a Metropolitan area network (MAN), a Wide area network (WAN), a Wireless wide area network (WWAN), enabled with technologies such as, by way of example, Global System for Mobile Communications (GSM), Personal Communications Service (PCS), Digital Advanced Mobile Phone Service (D-Amps), Bluetooth, Wi-Fi, Fixed Wireless Data, 2G, 2.5G, 3G, 4G, IMT-Advanced, pre-4G, 3G LTE, 3GPP LTE, LTE Advanced, mobile WiMax, WiMax 2, WirelessMAN-Advanced networks, enhanced data rates for GSM evolution (EDGE), General packet radio service (GPRS), enhanced GPRS, iBurst, UMTS, HSPDA, HSUPA, HSPA, UMTS-TDD, 1xRTT, EV-DO, messaging protocols such as, TCP/IP, SMS, MMS, extensible messaging and presence protocol (XMPP), real time messaging protocol (RTMP), instant messaging and presence protocol (IMPP), instant messaging, USSD, IRC, or any other wireless data networks or messaging protocols.

Third-party applications **120** may be provided by third-party software application vendors and may be accessible to the users via the network **106** (e.g., from servers hosted by the third-party software application vendors) or via, after download and installation, software codes (e.g., applications) running on the user devices **102**. In either configuration, the third-party applications **120** may communicate with the host server **110** for accessing cloud-based collaboration platform, storage and/or services in performing their functions, such as the document rendition techniques further discussed herein.

The embodiments disclosed herein recognize that the requirement of installing and executing a software (e.g., a plug-in) that is foreign to a user's system places a barrier to easy access documents that are in proprietary format (e.g., Adobe PDF, Microsoft Document DOC, Microsoft PowerPoint PPT, etc.). As such, it would be desirable to provide an intuitive way to view and interact with documents, thereby creating a seamless experience for sharing and collaborating on any or almost any browser or device.

Accordingly, the present disclosure enables a user to upload a document file (e.g., a PDF file) (e.g., via web application interface **104** or native client user interface **107** for it to be stored on a shared workspace in the cloud collaboration platform, or via a third-party application **120** using an application programming interface (API) for conversion or other suitable purposes), and receive a version of the same document in the user's browser or the third-party software running on the user's device **102**, which the user can then share with collaborators **108**. Further, the user can annotate, make notes or comments, highlight, fill text into blanks, as well as perform other actions to the document that can show up to other users (e.g., collaborators **108**) in real time.

More specifically, in some embodiments, the host server **110** can render documents to web browsers on devices **102**, to user interfaces **104** and **107**, and/or to third-party application **120** without requiring plug-ins, downloads, or software that would otherwise be necessary to view or access these documents. The host server **110** can also render the documents without compromising fidelity or performance.

In one or more embodiments, the host server **110** can render a document that is in a proprietary format (e.g., a PDF) by extracting all embedded resources (e.g., fonts, images, etc.) in the document. The host server **110** can convert a plurality of resources in a document file into a plurality of files that are native to a browser. In other words, the host server **110** can selectively convert the embedded resources so that all resources are in web-compatible formats, such as a PNG file for an image, or a TTF file for a font.

It is noted that the present disclosed techniques are not limited to rendering proprietary documents in a web browser; rather, the present techniques utilize the file formats that are native to a web browser so as to minimize the prerequisite of having a third-party plug-in or software installed on devices **102** before becoming able to view or render the proprietary documents.

Further, the host server **110** can create a style sheet based on the document file, so that an aggregate of the plurality of files together with the style sheet can cause the browser on device **102** or the third-party application **120** to render an appearance of the document file. More specifically, the embedded resources such as fonts that are shared between pages can be embedded within a shared web style sheet. The contents of the document file can be interpreted by the host server **110**. In some embodiments, as the document file is processed, graphic operations in the document file can be mapped to their equivalent scalable vector graphic (SVG) operations. Additional layout analysis can also be performed by the host server **110** when processing text in the document file to enable compact representation of text using SVG content.

In accordance with one or more embodiments, the host server **110** can further process the text to enable generation of an invisible layer, based on the document file, to be laid on the rendered appearance (or rendition) of the document file. The invisible layer can enable actions including, for example, text selecting, text copying, text cutting, text pasting, text searching, text filling, or hyperlinking, to be performed on the document file. In particular examples of the present disclosure, the host server **110** can use HyperText Markup Language (HTML) (e.g., HTML5) pages to render invisible text so as to form the invisible layer, and to enable text selection, hyperlinking, search, and other functionalities.

Among others, the advantages provided by the techniques disclosed herein are particularly beneficial for large scale and demand in consumer and business web and mobile applications. Using the techniques disclosed herein, proprietary documents (e.g., PDFs, PowerPoint or Word documents) can be embedded into almost any web or mobile application using a simple library that is native to most modern web browsers (e.g., iFrame or JavaScript), thus no plug-ins, downloads, or desktop software is required. The techniques can produce fast and high fidelity rendering, and in some embodiments, provide advanced security features including, for example, 256-bit document encryption, on-premise storage options, and multiple deployment options, including Software as a Service (SaaS) and private cloud.

More implementation details on the document rendering techniques which can be implemented on the host server **110** to render proprietary documents on web application interface

107, client software user interface 104, third-party application 120, or a third-party application running on the mobile client devices 102 are discussed in fuller detail below, and particularly with regard to FIG. 4.

FIG. 2 depicts an example diagram of a web-based or online collaboration platform deployed in an enterprise or other organizational setting 250 for organizing work items 215, 235, 255 and workspaces 205, 225, 245.

The web-based platform for collaborating on projects or jointly working on documents can be used by individual users and shared among collaborators. In addition, the collaboration platform can be deployed in an organized setting including but not limited to, a company (e.g., an enterprise setting), a department in a company, an academic institution, a department in an academic institution, a class or course setting, or any other types of organizations or organized setting.

When deployed in an organizational setting, multiple workspaces (e.g., workspace A, B C) can be created to support different projects or a variety of work flows. Each workspace can have its own associate work items. For example, workspace A 205 can be associated with work items 215, workspace B 225 can be associated with work items 235, and workspace N can be associated with work items 255. The work items 215, 235, and 255 can be unique to each workspace but need not be. For example, a particular word document can be associated with only one workspace (e.g., workspace A 205) or it can be associated with multiple workspaces (e.g., Workspace A 205 and workspace B 225, etc.).

In general, each workspace has a set of users or collaborators associated with it. For example, workspace A 205 is associated with multiple users or collaborators 206. In some instances, workspaces deployed in an enterprise can be department specific. For example, workspace B can be associated with department 210 and some users shown as example user A 208 and workspace N 245 can be associated with departments 212 and 216 and users shown as example user B 214.

Each user associated with a workspace can generally access the work items associated with the workspace. The level of access will depend on permissions associated with the specific workspace, and/or with a specific work item. Permissions can be set for the workspace or set individually on a per work item basis. For example, the creator of a workspace (e.g., one of user A 208 who creates workspace B) can set one permission setting applicable to all work items 235 for other associated users and/or users associated with the affiliate department 210, for example. Creator user A 208 can also set different permission settings for each work item, which can be the same for different users, or varying for different users.

In each workspace A, B . . . N, when an action is performed on a work item by a given user or any other activity is detected in the workspace, other users in the same workspace can be notified (e.g., in real time or in near real time, or not in real time). Activities which trigger real time notifications can include, by way of example but not limitation, adding, deleting, or modifying collaborators in the workspace, uploading, downloading, adding, deleting a work item in the workspace, creating a discussion topic in the workspace.

In some embodiments, items or content downloaded or edited can cause notifications to be generated. Such notifications can be sent to relevant users to notify them of actions surrounding a download, an edit, a change, a modification, a new file, a conflicting version, an upload of an edited or modified file.

In one embodiment, in a user interface to the web-based collaboration platform where notifications are presented, users can, via the same interface, create action items (e.g.,

tasks) and delegate the action items to other users including collaborators pertaining to a work item 215, for example. The collaborators 206 can be in the same workspace A 205 or the user can include a newly invited collaborator. Similarly, in the same user interface where discussion topics can be created in a workspace (e.g., workspace A, B or N, etc.), actionable events on work items can be created and/or delegated/assigned to other users such as collaborators of a given workspace 206 or other users. Through the same user interface, task status and updates from multiple users or collaborators can be indicated and reflected. In some instances, the users can perform the tasks (e.g., review or approve or reject, etc.) via the same user interface.

FIG. 3 depicts an example diagram of a workspace 302 in an online or web-based collaboration environment accessible by multiple collaborators 322 through various devices.

Each of users 316, 318, and 320 can individually use multiple different devices to access and/or manipulate work items 324 in the workspace 302 with which they are associated with. For example users 316, 318, 320 can be collaborators on a project to which work items 324 are relevant. Since the work items 324 are hosted by the collaboration environment (e.g., a cloud-based environment), each user can access the work items 324 anytime, and from any physical location using any device (e.g., including devices they own or any shared/public/loaner device).

Work items to be edited or viewed can be accessed from the workspace 302. Users can also be notified of access, edit, modification, and/or upload related-actions performed on work items 324 by other users or any other types of activities detected in the workspace 302. For example, if user 316 modifies a document, one or both of the other collaborators 318 and 320 can be notified of the modification in real time, or near real-time, or not in real time. The notifications can be sent through any of all of the devices associated with a given user, in various formats including, one or more of, email, SMS, or via a pop-up window in a user interface in which the user uses to access the collaboration platform. In the event of multiple notifications, each notification can be depicted preferentially (e.g., ordering in the user interface) based on user preferences and/or relevance to the user (e.g., implicit or explicit).

For example, a notification of a download, access, read, write, edit, or uploaded related activities can be presented in a feed stream among other notifications through a user interface on the user device according to relevancy to the user determined based on current or recent activity of the user in the web-based collaboration environment.

In one embodiment, the notification feed stream further enables users to create or generate actionable events (e.g., as task) which are or can be performed by other users 316 or collaborators 322 (e.g., including admin users or other users not in the same workspace), either in the same workspace 302 or in some other workspace. The actionable events such as tasks can also be assigned or delegated to other users via the same user interface.

For example, a given notification regarding a work item 324 can be associated with user interface features allowing a user 316 to assign a task related to the work item 324 (e.g., to another user 316, admin user 318, creator user 320 or another user). In one embodiment, a commenting user interface or a comment action associated with a notification can be used in conjunction with user interface features to enable task assignment, delegation, and/or management of the relevant work item or work items in the relevant workspaces, in the same user interface.

FIG. 4 depicts a block diagram illustrating an example of components in a host server (e.g., server **110**, FIG. 1) with capabilities that render documents in a browser (e.g., running on devices **102**), in a mobile device (e.g., through an application), or for a third-party application **120** regardless of whether a plug-in therefor is installed. With additional reference to FIGS. 1-3, the document rendition techniques which the host server **110** can employ are described.

The host server **110** of the web-based or online collaboration environment can generally be a cloud-based service. The host server **110** can include, for example, a network interface **410**, a document conversion engine **420** having a resource converter module **430**, a style sheet creator module **440**, an invisible layer (e.g., using an HTML page) generator module **450**, and a scalable vector graphic (SVG) output module **460**. In some embodiments of the host server **110**, the resource converter module **430** further includes a resource extractor/document analysis module **432**. One or more embodiments of the resource converter module **430** can also include, for example, a font processing module **434a**, an image processing module **434b**, a graphics processing module **434c**, and a text processing module **434d**.

The network interface **410** can be a networking module that enables the host server **110** to mediate data in a network with an entity that is external to the host server **110**, through any known and/or convenient communications protocol supported by the host and the external entity. The network interface **410** can include one or more of a network adaptor card, a wireless network interface card (e.g., SMS interface, WiFi interface, interfaces for various generations of mobile communication standards including but not limited to 1G, 2G, 3G, 3.5G, 4G, LTE, etc.), Bluetooth, a router, an access point, a wireless router, a switch, a multilayer switch, a protocol converter, a gateway, a bridge, bridge router, a hub, a digital media receiver, and/or a repeater.

As used herein, a “module,” “a manager,” an “interface,” or an “engine” includes a general purpose, dedicated or shared processor and, typically, firmware or software modules that are executed by the processor. Depending upon implementation-specific or other considerations, the module, manager, interface, or engine can be centralized or its functionality distributed. The module, manager, interface, or engine can include general or special purpose hardware, firmware, or software embodied in a computer-readable (storage) medium for execution by the processor. As used herein, a computer-readable medium or computer-readable storage medium is intended to include all media that are statutory (e.g., in the United States, under 35 U.S.C. §101), and to specifically exclude all media that are non-statutory in nature to the extent that the exclusion is necessary for a claim that includes the computer-readable (storage) medium to be valid. Known statutory computer-readable mediums include hardware (e.g., registers, random access memory (RAM), non-volatile (NV) storage, to name a few), but may or may not be limited to hardware.

As previously described, overall, the host server **400** can render documents in web browsers and mobile devices using open standards that are native to a browser (e.g., such as HyperText Markup Language (HTML), Scalable Vector Graphic (SVG), and Cascading Style Sheet (CSS)) without requiring third-party plugins such as “Adobe Flash.” In accordance with some of the disclosed embodiments, documents can be rendered without compromising fidelity or performance via delivering to the browser or to an application (e.g., via an application programming interface (API) such as API **415**) a mix of SVG and HTML. In addition, some embodiments of the host server **110** can stream pages to the browser

or the requesting application as they are generated or processed to enable instantaneous viewing of converted documents. Further, the host server **110** can generate additional HTML content to enable end-users to interact with documents in a variety of modes including, for example, collaborative annotation, form filling, electronic signing, document editing, searching, hyperlinking, and text selection (e.g., copying, etc.).

More specifically, a user of the host server **110** can submit a document in a proprietary format for rendering by one of several example ways, such as uploading the document (e.g., via a web interface **104**, or via a software client interface **107**) to be stored in the cloud workspace (e.g., in repository **130**). In some embodiments, a third-party application **120** (which can be running as a service on a third-party application server, or can be running as an application on devices **102**) can submit the document for rendering by “calling” or utilizing the API **415**.

The API **415** can include client-side APIs (e.g., for third-party applications running on devices **102**) and server-side APIs (e.g., for third-party applications running on third-party servers) that allow application developers to integrate the functionalities of the present embodiments into their own web applications and/or to customize for their uses. The API **415** can include functionality such as allowing users to upload documents, manage sharing settings, respond to document navigation events, and access annotation data. As such, it is noted that the present disclosed techniques are not limited to rendering proprietary documents in a web browser; rather, the present techniques utilize the file formats that are native to a web browser so as to minimize the prerequisite of having a third-party plug-in or software installed on devices **102** before becoming able to view or render the proprietary documents.

When the host server **110** receives a request to render the document (e.g., when a collaborator tries to view the document stored in the shared workspace on the collaborator’s computer in a web browser), the host server **110** can employ the document conversion engine **420** for the document rendition.

First, the host server **110** can utilize the resource converter module **430** to convert a plurality of resources in a document file into a plurality of files that are native to a browser. In other words, the host server **110** can selectively convert the embedded resources so that all resources are in web-compatible formats, such as a PNG file for an image, or a TTF file for a font. The resource extractor/document analysis module **432** can extract all embedded resources (e.g., fonts, images, etc.) in the document.

Further, the style sheet creator module **440** can create a style sheet based on the document file, so that an aggregate of the plurality of files together with the style sheet can cause the browser on device **102** or the third-party application **120** to render an appearance of the document file. More specifically, the embedded resources such as fonts that are shared between pages can be embedded within a shared web style sheet. The contents of the document file can be interpreted by the resource extractor/document analysis module **432**. In some embodiments, as the document file is processed, graphic operations in the document file can be mapped by the resource extractor/document analysis module **432**, together with the SVG output module **460**, to their equivalent scalable vector graphic (SVG) operations. Additional layout analysis can also be performed by the resource extractor/document analysis module **432** and the SVG output module **460** when processing text in the document file to enable compact representation of text using SVG content.

In accordance with one or more embodiments, the resource converter module **430** and the invisible layer generator module **450** can further process the text to enable generation of an invisible layer, based on the document file, to be laid on the rendered appearance (or rendition) of the document file. The invisible layer can enable actions including, for example, text selecting, text copying, text cutting, text pasting, text searching, text filling, or hyperlinking, to be performed on the document file. In particular examples of the present disclosure, the invisible layer generator module **450** can use HyperText Markup Language (HTML) (e.g., HTML5) pages to render invisible text so as to form the invisible layer, and to enable text selection, hyperlinking, search, and other functionalities.

Even more specifically, in some embodiments of the document conversion engine **420**, for font processing in the document, the embedded font can first be extracted by the resource extractor/layout analysis module **432**. Then, each glyph within the embedded font can be assigned a unique unicode value by the resource converter module **430** so that each glyph may easily be referenced within SVG text (e.g., as discussed below). In one embodiment, metadata for enabling crisper rendering on certain operating system (e.g., a Windows Operating System) can be calculated and added to the font file by the font processing module **434a**. The font can be saved using a format that is native to a web browser (e.g., a TTF font format) to enable use within web environments.

Moreover, a “blank” variation of each font can be created by the font processing module **434a** to enable use of invisible text within the HTML content (e.g., as generated by the invisible layer generator module **450**). It is noted that, in some implementations, the original fonts and the “blank” fonts may share identical font metrics; however, the glyphs within the “blank” fonts may contain no outline information.

In accordance with some embodiments of the document conversion engine **420**, for image processing in the document, embedded images are saved by the image processing module **434b** within their original formats to enable high-fidelity conversions when appropriate (e.g., when the original format is a web browser native format and/or when the original size/resolution is not too high as to severely impede the performance of document rendering, which may be a design choice by a person having ordinary skill in the art).

In some embodiments, the embedded images can be decoded and saved by the image processing module **434b** using a web-optimized image format (e.g., a PNG image format). Moreover, in some occasional instances where large images are embedded in the original document, the image processing module **434b** can down sample the original images to enable faster loading of content within the browser or the third-party application.

In accordance with some embodiments of the document conversion engine **420**, for graphics processing in the document, all graphics operations in the original document are mapped by the graphics processing module **434c** to their equivalent SVG operations as the document is processed by the document conversion engine **420**. According to one embodiment, the graphics processing module **434c** can employ nested objects to simulate layers, masks, patterns, and transparency groups within the SVG content. Depending on the document and the implementation, the graphics processing module **434c** can communicate with the style sheet creator module **440** so as to employ CSS properties to simulate the use of certain operations such as fill, strokes, and transparencies in the original document. All SVG contents are communicated to the SVG output module **460** which, together with the CSS contents created by the module **440**, cause the browser or the third-party application to render the

appearance of the document file regardless of whether a plug-in software that supports the document file is installed for the browser.

In accordance with some embodiments of the document conversion engine **420**, for text processing in the document, the resource extractor/document analysis module **432** can perform a layout analysis on the original document to enable the compact representation of text within the files as the pages of the original document are converted to, for example, HTML and SVG contents. The result of the text analysis can be used to, for example, group unstructured text within the original document (e.g., PDF) into words, lines, and text boxes. Afterwards, the text processing module **434d** can use the result of the text analysis, together with the SVG output module **460**, to render SVG text content. It is noted that, in one or more embodiments, text is rendered using SVG content instead of HTML content to enable higher-precision sub-pixel position of characters.

Moreover, the text processing module **434d** can use the result of the text analysis, together with the invisible layer generator module **450**, to create invisible text (e.g., on an invisible HTML layer). According to embodiments, the invisible HTML text can overlaid on top of the SVG content to enable actions to be performed on the document, such as text selection, hyperlinking, and search.

In some embodiments, to enable instantaneous loading of converted documents, pages are streamed to the browser as they are converted. In some of these embodiments, before any page is shown, a web style sheet containing shared fonts can be first loaded. Then, for each subsequent page, an SVG image can be used to display the contents of the page. Additionally, an invisible HTML layer can be overlaid on top of the SVG image to enable document interactivity. Besides those discussed above, further examples of document interactivity can include:

(a) collaborative annotation—which can allow users to add comments and other annotations to documents. In some examples, comments and annotations can be rendered using HTML in various layers over a document while being stored as separate metadata to facilitate being selectively shown/hidden as desired.

(b) form filling and electronic signatures—which can allow users to create fillable forms as well as fill out existing forms.

(c) document editing—which can allow the contents of document (e.g., PDF, DOC (or DOCX), and PPT (or PPTX)) to be edited directly through an edit interface. In some embodiments, the interface provides the so-called “What You See Is What You Get” (WYSIWYG) functionality.

It is noted that some documents (e.g., a PDF document) can sometimes contain a large quantity of embedded fonts and images. In some embodiments, to reduce the number of web requests that must be made before loading a page of a document, all or most unshared resources can be base64-embedded within the pages that contain them. Further, all shared fonts can be embedded within a single shared web style sheet, and in one or more embodiments, shared images are loaded as shared external resources.

In this way, the disclosed embodiments enable the users to view documents in their web browsers and mobile devices without having to download large files or use plug-ins or other software (e.g., a desktop software such as Adobe Acrobat). The embodiments can also enable a user to upload documents in proprietary formats (e.g., Word and PDF formats), and the user’s collaborators (including the user himself) can view, annotate, highlight, comment, as well as perform other suitable actions on the document without the need to download

the document files and open them using desktop software. Documents can also be loaded faster because, among other reasons, the payload that the browser or the third-party application has to load is reduced.

FIGS. 5A-5C respectively depict screenshots showing example user interfaces embodying one or more techniques disclosed herein for rendering documents in a browser or mobile device (e.g., device 102, FIG. 1) regardless of whether a plug-in therefor is installed. In FIG. 5A, a view interface 500 which can be used to render documents in, for example, a web browser, is shown.

In some embodiments, the view interface 500 can have a browsing bar 510 to allow the use to browse and select the documents stored in the shared workspace. An example of the browsing bar 510 is shown in FIG. 5B. The browsing bar 510 can be activated by, for example, a swipe down gesture performed by the user on a touchscreen of the mobile device 102. The browsing bar 510 can include miniature thumbnails, each thumbnail displaying an actual content of the document using the document rendering techniques disclosed herein, so that the browsing bar 510 of the interface 500 enables the user to take a quick glance at the actual content of the files without the need for the user to access the file. This can reduce time and effort for the user in reading and responding to the updates when collaborating with his or her collaborators on items shared on the workspace (e.g., workspace 302, FIG. 3).

The interface 500 can also include a collaboration bar 520 to allow the user to perform interactivity or collaboration functions on the document rendered (e.g., using the techniques disclosed herein). An example of the collaboration bar 520 is shown in FIG. 5C. The collaboration bar 520 can be activated by, for example, a swipe left gesture performed by the user on a touchscreen of the mobile device 102. The collaboration bar 510 can include actions that the user can perform on the viewed document, such as share, comment, download, or assign a task to another collaborator(s).

FIG. 6 depicts a flowchart illustrating an example process 600 for a host server (e.g., host server 110, FIGS. 1 and 4) in implementing the techniques disclosed herein for rendering documents (e.g., items 324 stored in workspace 302, FIG. 3; or received from a third-party application 120 via an application programming interface (API) 415, FIGS. 1 and 4) in a browser or mobile device (e.g., device 102, FIG. 1) regardless of whether a plug-in therefor is installed.

First, the host server 110 can convert (610) a plurality of resources in a document file into a plurality of files that are native to a browser. In other words, the host server 110 can selectively convert the embedded resources so that all resources are in web-compatible formats, such as a PNG file for an image, or a TTF file for a font. In some embodiments, the host server 110 can extract all embedded resources (e.g., fonts, images, etc.) in the document.

Further, the host server 110 can create (620) a style sheet based on the document file, so that an aggregate of the plurality of files together with the style sheet can cause (620) the browser on device 102 or the third-party application 120 to render an appearance of the document file. More specifically, the embedded resources such as fonts that are shared between pages can be embedded within a shared web style sheet. The contents of the document file can be interpreted by the host server 110. In some embodiments, as the document file is processed, graphic operations in the document file can be mapped by the host server 110 to their equivalent scalable vector graphic (SVG) operations. Additional layout analysis can also be performed by the host server 110 when processing text in the document file to enable compact representation of text using SVG content.

In accordance with one or more embodiments, the host server 110 can further process the text to generate (630) an invisible layer, based on the document file, to be laid on the rendered appearance (or rendition) of the document file. The invisible layer can enable (630) actions including, for example, text selecting, text copying, text cutting, text pasting, text searching, text filling, or hyperlinking, to be performed on the document file. In particular examples of the present disclosure, the host server 110 can use HyperText Markup Language (HTML) (e.g., HTML5) pages to render invisible text so as to form the invisible layer, and to enable text selection, hyperlinking, search, and other functionalities.

In addition, some embodiments of the host server 110 can stream (640) pages to the browser or the requesting application as they are generated or processed to enable instantaneous viewing of converted documents.

FIG. 7 shows a diagrammatic representation 700 of a machine in the example form of a computer system within which a set of instructions, for causing the machine to perform any one or more of the methodologies discussed herein, can be executed.

In alternative embodiments, the machine operates as a standalone device or can be connected (e.g., networked) to other machines. In a networked deployment, the machine can operate in the capacity of a server or a client machine in a client-server network environment, or as a peer machine in a peer-to-peer (or distributed) network environment.

The machine can be a server computer, a client computer, a personal computer (PC), a user device, a tablet, a phablet, a laptop computer, a set-top box (STB), a personal digital assistant (PDA), a thin-client device, a cellular telephone, an iPhone, an iPad, aBlackberry, a processor, a telephone, a web appliance, a network router, switch or bridge, a console, a hand-held console, a (hand-held) gaming device, a music player, any portable, mobile, hand-held device, or any machine capable of executing a set of instructions (sequential or otherwise) that specify actions to be taken by that machine.

While the machine-readable medium or machine-readable storage medium is shown in an exemplary embodiment to be a single medium, the term "machine-readable medium" and "machine-readable storage medium" should be taken to include a single medium or multiple media (e.g., a centralized or distributed database, and/or associated caches and servers) that store the one or more sets of instructions. The term "machine-readable medium" and "machine-readable storage medium" shall also be taken to include any medium that is capable of storing, encoding or carrying a set of instructions for execution by the machine and that cause the machine to perform any one or more of the methodologies of the presently disclosed technique and innovation.

In general, the routines executed to implement the embodiments of the disclosure, can be implemented as part of an operating system or a specific application, component, program, object, module or sequence of instructions referred to as "computer programs." The computer programs typically comprise one or more instructions set at various times in various memory and storage devices in a computer, and that, when read and executed by one or more processing units or processors in a computer, cause the computer to perform operations to execute elements involving the various aspects of the disclosure.

Moreover, while embodiments have been described in the context of fully functioning computers and computer systems, those skilled in the art will appreciate that the various embodiments are capable of being distributed as a program product in a variety of forms, and that the disclosure applies

equally regardless of the particular type of machine or computer-readable media used to actually effect the distribution.

Further examples of machine-readable storage media, machine-readable media, or computer-readable (storage) media include, but are not limited to, recordable type media such as volatile and non-volatile memory devices, floppy and other removable disks, hard disk drives, optical disks (e.g., Compact Disk Read-Only Memory (CD ROMS), Digital Versatile Disks, (DVDs), etc.), among others, and transmission type media such as digital and analog communication links.

The network interface device enables the machine 2800 to mediate data in a network with an entity that is external to the host server, through any known and/or convenient communications protocol supported by the host and the external entity. The network interface device can include one or more of a network adaptor card, a wireless network interface card, a router, an access point, a wireless router, a switch, a multi-layer switch, a protocol converter, a gateway, a bridge, bridge router, a hub, a digital media receiver, and/or a repeater.

The network interface device can include a firewall which can, in some embodiments, govern and/or manage permission to access/proxy data in a computer network, and track varying levels of trust between different machines and/or applications. The firewall can be any number of modules having any combination of hardware and/or software components able to enforce a predetermined set of access rights between a particular set of machines and applications, machines and machines, and/or applications and applications, for example, to regulate the flow of traffic and resource sharing between these varying entities. The firewall can additionally manage and/or have access to an access control list which details permissions including for example, the access and operation rights of an object by an individual, a machine, and/or an application, and the circumstances under which the permission rights stand.

Other network security functions can be performed or included in the functions of the firewall, can be, for example, but are not limited to, intrusion-prevention, intrusion detection, next-generation firewall, personal firewall, etc. without deviating from the novel art of this disclosure.

Unless the context clearly requires otherwise, throughout the description and the claims, the words “comprise,” “comprising,” and the like are to be construed in an inclusive sense, as opposed to an exclusive or exhaustive sense; that is to say, in the sense of “including, but not limited to.” As used herein, the terms “connected,” “coupled,” or any variant thereof, means any connection or coupling, either direct or indirect, between two or more elements; the coupling of connection between the elements can be physical, logical, or a combination thereof. Additionally, the words “herein,” “above,” “below,” and words of similar import, when used in this application, shall refer to this application as a whole and not to any particular portions of this application. Where the context permits, words in the above Detailed Description using the singular or plural number can also include the plural or singular number respectively. The word “or,” in reference to a list of two or more items, covers all of the following interpretations of the word: any of the items in the list, all of the items in the list, and any combination of the items in the list.

The above detailed description of embodiments of the disclosure is not intended to be exhaustive or to limit the teachings to the precise form disclosed above. While specific embodiments of, and examples for, the disclosure are described above for illustrative purposes, various equivalent modifications are possible within the scope of the disclosure, as those skilled in the relevant art will recognize. For example, while processes or blocks are presented in a given

order, alternative embodiments can perform routines having steps, or employ systems having blocks, in a different order, and some processes or blocks can be deleted, moved, added, subdivided, combined, and/or modified to provide alternative or subcombinations. Each of these processes or blocks can be implemented in a variety of different ways. Also, while processes or blocks are at times shown as being performed in series, these processes or blocks can instead be performed in parallel, or can be performed at different times. Further, any specific numbers noted herein are only examples: alternative implementations can employ differing values or ranges.

The teachings of the disclosure provided herein can be applied to other systems, not necessarily the system described above. The elements and acts of the various embodiments described above can be combined to provide further embodiments.

Any patents and applications and other references noted above, including any that can be listed in accompanying filing papers, are incorporated herein by reference. Aspects of the disclosure can be modified, if necessary, to employ the systems, functions, and concepts of the various references described above to provide yet further embodiments of the disclosure.

These and other changes can be made to the disclosure in light of the above Detailed Description. While the above description describes certain embodiments of the disclosure, and describes the best mode contemplated, no matter how detailed the above appears in text, the teachings can be practiced in many ways. Details of the system can vary considerably in its implementation details, while still being encompassed by the subject matter disclosed herein. As noted above, particular terminology used when describing certain features or aspects of the disclosure should not be taken to imply that the terminology is being redefined herein to be restricted to any specific characteristics, features, or aspects of the disclosure with which that terminology is associated. In general, the terms used in the following claims should not be construed to limit the disclosure to the specific embodiments disclosed in the specification, unless the above Detailed Description section explicitly defines such terms. Accordingly, the actual scope of the disclosure encompasses not only the disclosed embodiments, but also all equivalent ways of practicing or implementing the disclosure under the claims.

While certain aspects of the disclosure are presented below in certain claim forms, the inventors contemplate the various aspects of the disclosure in any number of claim forms. For example, while only one aspect of the disclosure is recited as a means-plus-function claim under 35 U.S.C. § 112, ¶6, other aspects can likewise be embodied as a means-plus-function claim, or in other forms, such as being embodied in a computer-readable medium. (Any claim intended to be treated under 35 U.S.C. § 112, ¶6 begins with the words “means for”.) Accordingly, the applicant reserves the right to add additional claims after filing the application to pursue such additional claim forms for other aspects of the disclosure.

What is claimed is:

1. A method for rendering a document, the method comprising:
 - converting a plurality of resources in a document file into a plurality of files that are native to a browser;
 - creating a style sheet based on the document file, wherein an aggregate of the plurality of files together with the style sheet are configured to cause the browser to render an appearance of the document file; and
 - generating, based on the document file, an invisible layer to be laid on the appearance, wherein the invisible layer enables actions to be performed on the document file.

17

2. The method of claim 1, further comprising:
streaming pages of the document file to the browser individually as each page is processed.
3. The method of claim 1, wherein the plurality of resources comprise a font, the method further comprising:
extracting the font from the document file; and
creating a blank variation of the font so as to enable the generation of the invisible layer, wherein glyphs within the blank variation of the font contain no outline information.
4. The method of claim 1, wherein the plurality of resources comprise an image, the method further comprising:
extracting the image from the document file; and
down-sampling the image if a size of the image is larger than a predetermined number.
5. The method of claim 1, wherein the plurality of resources comprise a graphic, the method further comprising:
mapping the graphic to an equivalent scalable vector graphic operation.
6. The method of claim 1, wherein the plurality of resources comprise text, the method further comprising:
mapping the text using to a plurality of equivalent scalable vector graphic operations; and
providing the text for the generation of the invisible layer.
7. The method of claim 1, wherein the document file is received from a third-party software application via an application programming interface (API).
8. The method of claim 1, wherein the browser is to render the appearance of the document file regardless of whether a plug-in software that supports the document file is installed for the browser.
9. The method of claim 1, wherein the invisible layer comprises a HyperText Markup Language (HTML) page.
10. The method of claim 1, wherein the actions to be performed on the document file includes one or more of text selecting, text copying, text cutting, text pasting, text searching, text filling, and hyperlinking.
11. The method of claim 1, wherein a file type of the document file includes one or more of Portable Document Format file (PDF), Word Document file (DOC), PowerPoint Presentation file (PPT), Excel worksheet file (XLS).
12. The method of claim 1, wherein the rendering is performed by a server which hosts a cloud-based environment shared among a user and collaborators of the user.
13. The method of claim 12, wherein the cloud-based environment includes a cloud-based collaboration environment, a cloud-based storage service or file sharing service.
14. A computer server which hosts a cloud-based environment shared among a user and collaborators of the user, the server comprising:
a processor; and
a memory unit having instructions stored thereon which when executed by the processor, causes the processor to:
convert a plurality of resources in a document file into a plurality of files that are native to a browser;
create a style sheet based on the document file, wherein an aggregate of the plurality of files together with the style sheet are configured to cause the browser to render an appearance of the document file; and
generate, based on the document file, an invisible layer to be laid on the appearance, wherein the invisible layer enables actions to be performed on the document file.
15. The server of claim 14, wherein the processor is further caused to:
stream pages of the document file to the browser individually as each page is processed.

18

16. The server of claim 14, wherein the plurality of resources comprise a font, and wherein the processor is further caused to:
extract the font from the document file; and
create a blank variation of the font so as to enable the generation of the invisible layer, wherein glyphs within the blank variation of the font contain no outline information.
17. The server of claim 14, wherein the plurality of resources comprise an image, and wherein the processor is further caused to:
extract the image from the document file; and
down-sample the image if a size of the image is larger than a predetermined number.
18. The server of claim 14, wherein the plurality of resources comprise a graphic, and wherein the processor is further caused to:
map the graphic to an equivalent scalable vector graphic operation.
19. The server of claim 14, wherein the plurality of resources comprise text, and wherein the processor is further caused to:
map the text using to a plurality of equivalent scalable vector graphic operations; and
provide the text for the generation of the invisible layer.
20. The server of claim 14, wherein the document file is received from a third-party software application via an application programming interface (API).
21. The server of claim 14, wherein the browser is to render the appearance of the document file regardless of whether a plug-in software that supports the document file is installed for the browser.
22. The server of claim 14, wherein the invisible layer comprises a HyperText Markup Language (HTML) page.
23. A machine-readable storage medium having stored thereon instructions which, when executed by a processor on a server, cause the processor to:
convert a plurality of resources in a document file into a plurality of files that are native to a browser;
create a style sheet based on the document file, wherein an aggregate of the plurality of files together with the style sheet are configured to cause the browser to render an appearance of the document file; and
generate, based on the document file, an invisible layer to be laid on the appearance, wherein the invisible layer enables actions to be performed on the document file.
24. The medium of claim 23, wherein the processor is further caused to:
stream pages of the document file to the browser individually as each page is processed.
25. The medium of claim 23, wherein the plurality of resources comprise a font, and wherein the processor is further caused to:
extract the font from the document file; and
create a blank variation of the font so as to enable the generation of the invisible layer, wherein glyphs within the blank variation of the font contain no outline information.
26. The medium of claim 23, wherein the plurality of resources comprise an image, and wherein the processor is further caused to:
extract the image from the document file; and
down-sample the image if a size of the image is larger than a predetermined number.
27. The medium of claim 23, wherein the plurality of resources comprise a graphic, and wherein the processor is further caused to:

map the graphic to an equivalent scalable vector graphic operation.

28. The medium of claim 23, wherein the plurality of resources comprise text, and wherein the processor is further caused to:

map the text using to a plurality of equivalent scalable vector graphic operations; and

provide the text for the generation of the invisible layer.

29. The medium of claim 23, wherein the document file is received from a third-party software application via an application programming interface (API); wherein the browser is to render the appearance of the document file regardless of whether a plug-in software that supports the document file is installed for the browser.

30. A system which hosts a cloud-based environment shared among a user and collaborators of the user, comprising:

means for converting a plurality of resources in a document file into a plurality of files that are native to a browser;

means for creating a style sheet based on the document file, wherein an aggregate of the plurality of files together with the style sheet are configured to cause the browser to render an appearance of the document file;

means for generating, based on the document file, an invisible layer to be laid on the appearance, wherein the invisible layer enables actions to be performed on the document file; and

means for streaming pages of the document file to the browser individually as each page is processed.

* * * * *

5

10

15

20

25

30